

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-090031

(43)Date of publication of application : 31.03.2000

(51)Int.Cl. G06F 13/00
H04L 12/24
H04L 12/26
H04L 12/56

(21)Application number : 10-257363

(71)Applicant : ISHII TAKESHI
NATL AEROSPACE LAB
FUJITA NAOYUKI

(22)Date of filing : 11.09.1998

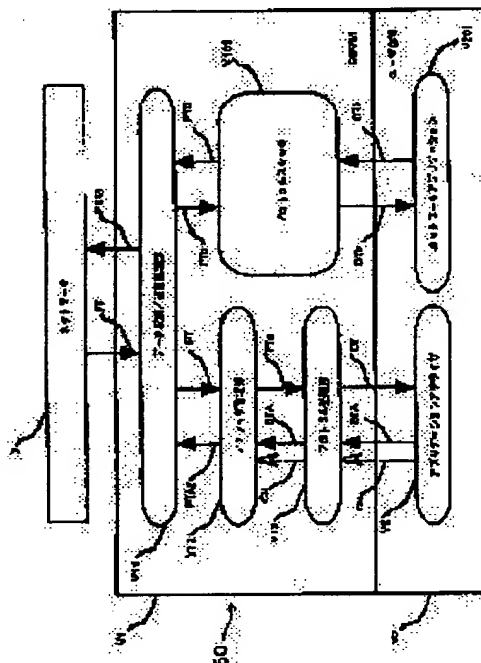
(72)Inventor : ISHII TAKESHI
FUJITA NAOYUKI

(54) METHOD FOR MONITORING/CONTROLLING NETWORK COMMUNICATION, MONITORING CONTROLLER USING THE SAME AND COMPUTER READABLE RECORDING MEDIUM WHICH RECORDS MONITORING/CONTROLLING PROGRAM OF NETWORK COMMUNICATION

(57)Abstract:

PROBLEM TO BE SOLVED: To easily deal with an arbitrary data system.

SOLUTION: The monitoring/controller of network communication is provided with a data collection/transmission processing part V11 transmitting/receiving packets (PT and PTA1) to/from a network 3, an analyzer managing part V12 provided with a hijack function which executes packet communication with the data collection/transmission processing part V11 based on analyzer managing information rewritten by a control message CM and which individually communicates with communicating persons by the packets (PT and PTA2) when injustice is detected in communication, a protocol processing part V13 reconstituting analysis data DT from the packet PTa and transmitting the control message CM and data DTA and an application analyzer V21 judging whether injustice does not exist in analysis data DT which is taken in and transmitting the control message CM and data DTA in accordance with the judged result.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-90031

(P2000-90031A)

(43) 公開日 平成12年3月31日 (2000.3.31)

| (51) Int.Cl. ⁷ | 識別記号 | F I | テーマコード(参考) |
|---------------------------|-------|---------------|-------------------|
| G 0 6 F 13/00 | 3 5 3 | G 0 6 F 13/00 | 3 5 3 U 5 B 0 8 9 |
| H 0 4 L 12/24 | | H 0 4 L 11/08 | 5 K 0 3 0 |
| 12/26 | | 11/20 | 1 0 2 Z |
| 12/56 | | | |

審査請求 未請求 請求項の数 8 O L (全 32 頁)

(21) 出願番号 特願平10-257363

(22) 出願日 平成10年9月11日 (1998.9.11)

(71) 出願人 598124939

石井 剛

東京都調布市染地3-9-51

(71) 出願人 391037397

科学技術庁航空宇宙技術研究所長

東京都調布市深大寺東町7丁目44番地1

(71) 出願人 597033812

藤田 直行

東京都小金井市市前原町4-17-28

(72) 発明者 石井 剛

東京都調布市染地3-9-51

(74) 代理人 100087468

弁理士 村瀬 一美

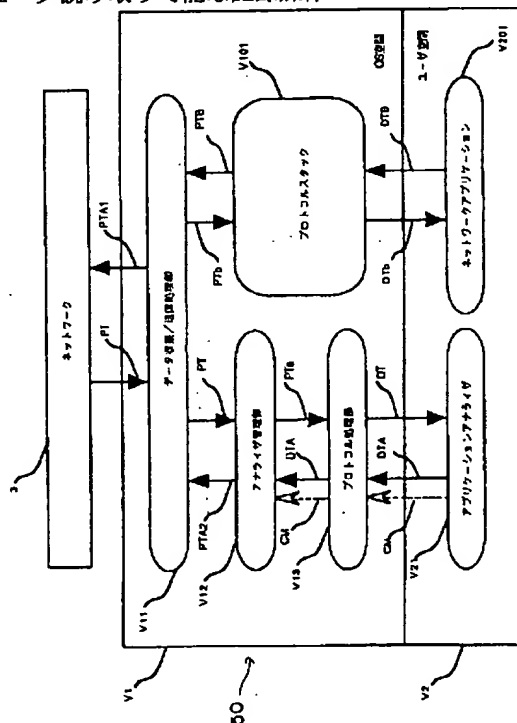
最終頁に続く

(54) 【発明の名称】 ネットワーク通信の監視・制御方法及びこれを利用した監視・制御装置並びにネットワーク通信の監視・制御プログラムを記録したコンピュータ読み取り可能な記録媒体

(57) 【要約】

【課題】 任意のデータ形式を簡便に取り扱う。

【解決手段】 ネットワーク通信の監視・制御装置は、ネットワーク3に対してパケット (PT, PTA1) を送受信できるデータ収集/送信処理部V11 と、制御メッセージCMによって書き換えられるアナライザ管理情報を基にデータ収集/送信処理部V11 との間でパケット通信を行い、通信に不正を検知したときパケット (PT, PTA2) でもって通信当事者と個々に通信するハイジャック機能を備えたアナライザ管理部V12 と、パケットPTaから解析データDTを再構築し、制御メッセージCM及びデータDTAを送出するプロトコル処理部V13 と、取り込んだ解析データDTに不正がないか判断し、その判断結果に応じて制御メッセージCM及びデータDTAを送出するアプリケーションアナライザV21 とを備える。



【特許請求の範囲】

【請求項1】 情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析する方法であって、前記ネットワークに対してパケットを送受信できるデータ収集/送信処理工程と、制御メッセージによって書き換えられるアナライザ管理情報を基に前記データ収集/送信処理工程との間でパケット通信を行うとともに、通信に不正が検知されたときには前記パケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理工程と、前記アナライザ管理工程から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータを前記アナライザ管理工程に伝送するプロトコル処理工程と、前記プロトコル処理工程から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理工程に送出するアナライザ工程とを備えたことを特徴とするネットワーク通信の監視・制御方法。

【請求項2】 前記データ収集/送信処理工程からのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにして前記データ収集/送信処理工程に与えるデータ作成送出工程と、前記データ作成送出工程から解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データを前記データ作成送出工程に与えるネットワーク処理工程とを備えたことを特徴とする請求項1に記載のネットワーク通信の監視・制御方法。

【請求項3】 前記データ収集/送信処理工程、アナライザ管理工程、プロトコル処理工程及びデータ作成送出工程、または前記データ収集/送信処理工程及びデータ作成送出工程がオペレーティングシステム空間に実装されたものであることを特徴とする請求項1または2記載のネットワーク通信の監視・制御方法。

【請求項4】 情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析する装置であって、前記ネットワークに対してパケットを送受信できるデータ収集/送信処理部と、制御メッセージによって書き換えられるアナライザ管理情報を基に前記データ収集/送信処理部との間でパケット通信を行うとともに、通信に不正が検知されたときには前記パケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理部と、前記アナライザ管理部から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータを前記アナライザ管理部に伝送するプロトコル処理部と、前記プロトコル処理部から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理部に送出するアプ

リケーションアナライザとを備えたことを特徴とするネットワーク通信の監視・制御装置。

【請求項5】 前記データ収集/送信処理部からのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにして前記データ収集/送信処理部に与えるプロトコルスタックと、前記プロトコルスタックから解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データを前記プロトコルスタックに与えるネットワークアプリケーションとを備えたことを特徴とする請求項4に記載のネットワーク通信の監視・制御装置。

【請求項6】 前記データ収集/送信処理部、アナライザ管理部、プロトコル処理部及びプロトコルスタック、または前記データ収集/送信処理部及びプロトコルスタックがオペレーティングシステム管理空間に実装されたものであることを特徴とする請求項4または5記載のネットワーク通信の監視・制御装置。

【請求項7】 情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析する方法であって、前記ネットワークに対してパケットを送受信できるデータ収集/送信処理手順と、制御メッセージによって書き換えられるアナライザ管理情報を基に前記データ収集/送信処理手順との間でパケット通信を行うとともに、通信に不正が検知されたときには前記パケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理手順と、前記アナライザ管理手順から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータを前記アナライザ管理手順に伝送するプロトコル処理手順と、前記プロトコル処理手順から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理手順に送出するアナライザ手順とをコンピュータに実行させるためのプログラムを記録したコンピュータ読み取り可能な記憶媒体。

【請求項8】 前記データ収集/送信処理手順からのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにして前記データ収集/送信処理手順に与えるデータ作成送出手順と、前記データ作成送出手順から解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データを前記データ作成送出手順に与えるネットワーク処理手順とをコンピュータに実行させるためのプログラムを記録した請求項7記載のコンピュータ読み取り可能な記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ネットワーク通信の監視・制御方法及びこれを利用した監視・制御装置並

びにネットワーク通信の監視・制御プログラムを記録したコンピュータ読み取り可能な記録媒体に関する。

【0002】さらに詳述すると、本発明は、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおいて、通信されるデータに不正が発生しているか否かを解析し、ネットワークのセキュリティやネットワーク管理を実行できるとともに、情報通信ステーションのオペレーティングシステムへのインターフェースとして実現されているネットワーク通信の監視・制御方法及びこれを利用した監視・制御装置並びにネットワーク通信の監視・制御プログラムを記録したコンピュータ読み取り可能な記録媒体に関するものである。

【0003】なお、本発明において、ネットワークにおける「不正」とは、例えばクラッキング等のいわゆる不正アクセスを意味する。

【0004】

【従来の技術】従来、情報通信ステーションの間で通信を行うように構築されたネットワークにおいて、その通信データに不正が発生しているか否かを監視する監視ツールが提供されている。

【0005】かかる従来のネットワーク監視ツールとしては、大別して、「パケット単位の監視ツール」と、「代理サーバ形式の監視ツール」とが挙げられる。

【0006】図38に、従来のパケット単位の監視ツールの構成例を示す。このパケット単位の監視ツールは、図38に示すように、ネットワークNTWで接続された通信装置MCNA、MCNBの間に通信装置MCNCを備え、この通信装置MCNCの内部で実現されるものである（第1の従来技術）。各通信装置MCNA、MCNBは、当該装置内のオペレーティングシステム（OS）空間にプロトコルスタックを、同ユーザ空間に通信ライブラリ及びアプリケーションを備えている。また、通信装置MCNCは、当該装置内のOS空間にプロトコルスタックを、同ユーザ空間に通信ライブラリ及び従来のアナライザを備えている。

【0007】このような両通信装置MCNA、MCNBの間で通信を行っている場合に、ネットワークNTW上を流れるパケット（通信内容[Hello B!]、通信相手[A->B]）PTを通信装置MCNCがそのまま受け取り、各種プロトコルを通信装置MCNCで実現したツール（従来のアナライザ）自らの責任で監視するものである。この種類のツールは作成が容易で数多く開発されている（Sniffer, Real Secure, NetRanger）。

【0008】図39に、従来の代理サーバ形式の監視ツールの構成例を示す。この代理サーバ形式の監視ツールは、図39に示すように、ネットワークNTWで接続された通信装置MCNA、MCNBの間に通信装置MCNCを備え、この通信装置MCNCが実際のサービスを中継することによって通信内容を傍受し解析することによ

り実現されるものである（第2の従来技術）。両通信装置MCNA、MCNBは、当該装置内のOS空間にプロトコルスタックを、同ユーザ空間に通信ライブラリ及びアプリケーションを備えている。また、通信装置MCNCは、当該装置内のOS空間にプロトコルスタックを、同ユーザ空間に通信ライブラリ及び代理サーバを備えている。

【0009】このような両通信装置MCNA、MCNBの間で通信を行っている場合に、例えば通信装置MCNAはネットワークNTW上にパケット（通信内容[Hello B!]、通信相手[A->C]）PTを送出する。すると、通信装置MCNCは、ウィルススキャナのように、プロトコルスタック及び通信ライブラリを介して当該パケットPTから通信内容[Hello B!]を受け取って当該通信内容を代理サーバが傍受し解析し、問題なければ、再び、当該通信内容を通信ライブラリ及びプロトコルスタックに渡して、この通信内容に通信相手[C->B]を付加してネットワークNTWを介して通信装置MCNBに渡すようにしたものである。この監視ツールによれば、TCPのようなコネクション指向の通信を正確に監視できるという利点がある（ServerProtect, VirusBuster）。

【0010】

【発明が解決しようとする課題】しかしながら、上記第1の従来技術によれば、ツール自体が多くのプロトコルを処理する基本設計になっているためツール自体が巨大化すること、パケット単位のデータをツール自体の責任で処理するためユーザが新しいものを作り出すのが難しいこと、等の欠点があった。

【0011】また、この第1の従来技術によれば、TCP（Transport Control Protocol）のようなコネクション指向のプロトコルを処理するには、ツールの中でマルチタスク化等を行う必要があり、ユーザに高度なプログラミング技術を要求することになる。このため、多くの場合、この種のツールはパケット単位の監視のみを行うこととなり、セッション単位の監視を行うことができないという欠点があった。

【0012】また、上記第2の従来技術によれば、サーバ及びクライアントの環境が代理サーバの存在を前提に構築されている必要があること、パケット単位の監視は不可能であること、などの問題点がある。

【0013】そこで、本発明の第1の目的は、アプリケーションのセッションを含め任意のデータ形式を簡便に取り扱うネットワーク通信の監視・制御方法及びこれを利用した監視・制御装置並びにネットワーク通信の監視・制御プログラムを記録したコンピュータ読み取り可能な記録媒体を提供することにある。

【0014】また、本発明の第2の目的は、データの単位を限定することなく、かつ、ネットワーク通信における既存機器に変更を加えることなく、簡便にネットワークの監視・制御を行う方法及び装置並びにその制御プロ

グラムを記憶した記憶媒体を提供することにある。

【0015】さらに、本発明の第3の目的は、簡便に利用できるユーザインターフェースを提供することにある。

【0016】

【課題を解決するための手段】本発明者らは、従来の監視ツールについて検討した結果、オペレーティングシステムがネットワークの監視・制御ツールの存在を予期することなく設計されたことが上記欠点や問題点を招来させた原因であること、ネットワーク通信がオペレーティングシステムに組み込まれた機能を利用することを前提としており、オペレーティングシステムのサポートなしで当該ツールを開発するのは困難かつ非効率なことであることを見出し、これを次のようにすることによって解決した。

【0017】すなわち、多くのネットワーク通信の中からある特定のネットワーク通信を選択するためには、

- (i) 発信者の識別子群
 - (ii) 受信者の識別子群
 - (iii) 通信のトリガ
- を指定する必要がある。

【0018】しかしながら、現在のプロトコルスタックと通信ライブラリには、後述する制限が存在するため、この3種類の条件を全てを指定することはできない。以下には、その制限とこれを解決するための手段を挙げる。

【0019】第1に、プロトコルスタックと通信ライブラリに施す変更点について検討する。

【0020】<プロトコルスタックに対する変更点>プロトコルスタックは通常階層構造になっている。ネットワークの物理的な処理を行う層からアプリケーションに近い層まで、多くのプロトコル層が存在する。また、プロトコルスタックは通常同時に複数の通信を処理する。その際、ある通信と別の通信を区別する識別子が必要となる。各プロトコル層で通信のなんらかの識別子を1つまたは複数持つことが多く、その個別の識別子をID(m)と書き、識別子候補と呼ぶことにする。通信はサーバとクライアントの組みで行われるので、サーバ側の識別子候補をSID(m)、クライアント側の識別子候補をCID(m)と書くことにする。1つの通信はSID(m)とCID(m)の組みで表すことができる。これを通信の識別子と呼びIと書くことにする。

【0021】ここで、サーバ側のプロトコルスタックに注目すると、通信の識別子Iは数式1のように記述できる。

【0022】

【数1】 $S = (SID(0), \dots, SID(s))$
 $C = (CID(0), \dots, CID(c))$

但し(すべてのmについて)(SID(m)はサーバのアドレスを表さない)。

【0023】この条件のもとで、数式2のようになる。

【0024】

【数2】 $I = S \wedge C$

そして、SID(m)の中でサーバ自身を表すアドレスのような値は除外される。なぜなら、サーバ自身が通信しているということは自明であるからである。クライアントのプロトコルスタックも同様に数式3のように表される。

【0025】

【数3】 $S = (SID(0), \dots, SID(s))$
 $C = (CID(0), \dots, CID(c))$

但し(すべてのmについて)(CID(m)はクライアントのアドレスを表さない)

$I = S \wedge C$

【0026】さて、ネットワーク解析ツールのためのプロトコルスタックを考えた場合、プロトコルスタックを実装しているホストが通信の当事者であるという前提を取り除かなければならない。すると通信識別子I'を数式4のように表すことができる。

【0027】

【数4】

$S' = (SID(0), \dots, SID(s))$
 $C' = (CID(0), \dots, CID(c))$

但し(あるmについて)(SID(m)はサーバのアドレスを表す)

(あるmについて)(CID(m)はクライアントのアドレスを表す)

さらに、このような条件のもとで、数式5のように表現することができる。

【0028】

【数5】 $I' = S' \wedge C'$

【0029】<通信ライブラリに対する変更点>従来の通信ライブラリは必ず通信の当事者用に設計されていた。サーバであるならば、サーバとして行うサービスアドレスの指定、クライアントであるならば、サーバのアドレス等を指定できるよう通信ライブラリが作られている。

【0030】すなわち、func(アドレス指定) ここで、アドレスはサーバまたはクライアントを特定するものである。

【0031】クライアントがサーバにアクセスするタイミングを持って通信開始とする。つまり、そのタイミングをトリガとして扱っている。

【0032】ここで、ネットワーク監視ツールのためのライブラリはどのような変更が必要になるかを考えると、ネットワーク監視ツールは通信の非当事者のため通信開始というタイミングがない。そこで、監視対象の通信が開始された時点をもってネットワーク監視ツールもアクティブになればよい。

【0033】しかしながら、もっと一般的にネットワーク通信を表すなんらかのトリガがありさえすればよい。

通信の非当事者のため必ずしも通信開始をトリガにする必要はない。言い換えれば、従来のアドレス指定は、下記のようにトリガ指定の中に含まれて記述できるようになる。

【0034】func(トリガ指定)

例えばトリガとしては次のようなものである。

【0035】・CID(0)=123 かつSID(5)=456の通信

・SID(0)=1かつSID(1)=2かつSID(3)=3の通信

・全ての通信

【0036】第2に不正な通信に対する制御方法について検討する。これは、不正な通信が行われているときに、単に、不正が存在するとするだけでなく、この不正に積極的に関与して不正な通信がされないようにするものである。以下、その不正な通信をさせない一つの対策である「通信のハイジャック機能」について検討する。

【0037】＜通信のハイジャックについて＞ネットワーク通信は通常サーバとクライアントの間で行われる。通信経路の途中でその通信を切断し、切断者がサーバとクライアントとのそれぞれと通信を行う。サーバはクライアントと、クライアントはサーバと通信しているつもりであるが、これらの実際の通信相手は切断者である。これを通信のハイジャックと呼ぶことにする。従来から通信のハイジャックは行われてきた。しかしながら、それは、一時的にまたは数パケットのみのハイジャックであったり、あるいは通信のタイミングに厳密性が必要であって多くの試行をくり返した後ハイジャックに成功するというようなものであった。

【0038】すなわち、このような従来のハイジャック技術では、ネットワーク侵入者に対して、攻撃的な対応をとることは不可能であり、通信の不正の発覚時に管理者が通信を切断するというあくまで防御の手段しか取りえなかった。

【0039】本発明において実現する通信のハイジャック機能は侵入者に対して初めて攻撃的な対応を管理者に与えるものである。

【0040】＜ハイジャックのために必要な変更＞通信の非当事者用のプロトコルスタックと通信ライブラリをデュアルホームホストに適用すると、通信のハイジャックが可能になる。以下、具体的な変更点について説明する。

【0041】＜プロトコルスタックに対する変更＞プロトコルスタックに対する変更については、以下のようなものがある。

【0042】・2インターフェース対応

・パケットのフォワード

・パケットのフォワード禁止ルール

・アプリケーションからの要求によるパケットの作成

【0043】＜通信ライブラリに対する追加＞通信ライブラリに対して追加する項目は、ハイジャックを行うための関数及びハイジャック後の通信を行うための関数と

が挙げられる。

【0044】デュアルホームホストの片方に実際の通信のサーバ、他方に実際の通信のクライアントがあると、デュアルホームホストにてパケットのフォワードを行っている場合を想定する。さらに、このホスト上でネットワーク解析ツールがサーバとクライアントの通信を解析しているとする。このときネットワーク解析ツールはサーバとクライアントの通信遮断を要求することができる（通信ライブラリに対する追加機能による）。デュアルホームホストのプロトコルスタックはパケットフォワード禁止ルールに要求のあった通信を登録する。この時点でサーバとクライアントの通信は切断される。次にネットワーク解析ツールはサーバと対話することができる（プロトコルスタックのパケット作成機能による）。

【0045】サーバは依然としてクライアントと通信を続けているようで、実はネットワーク解析ツールと対話をしているのである。クライアントに対しても同様である。

【0046】上述した検討に基づいて想起された請求項1記載の発明に係るネットワーク通信の監視・制御方法は、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析する方法であって、ネットワークに対してパケットを送受信できるデータ収集／送信処理工程と、制御メッセージによって書き換えられるアナライザ管理情報を基にデータ収集／送信処理工程との間でパケット通信を行うとともに、通信に不正が検知されたときにはパケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理工程と、アナライザ管理工程から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータをアナライザ管理工程に伝送するプロトコル処理工程と、プロトコル処理工程から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理工程に送出するアナライザ工程とを備えたことを特徴とする。

【0047】したがって、パケットから解析データを再構築して解析できるので、再構築後のデータにより不正の判断を行うことができる。また、デュアルホームホストの採用によりハイジャック機能を実現できるので、ネットワークでの不正を防止することができる。

【0048】また、請求項2記載の発明は、請求項1において、データ収集／送信処理工程からのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにしてデータ収集／送信処理工程に与えるデータ作成送出工程と、データ作成送出工程から解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データをデータ作成送出工程に与えるネットワーク処理工程とを備えたことを特徴

とする。

【0049】さらに、請求項3記載の発明は、請求項1または2において、データ収集/送信処理工程、アナライザ管理工程、プロトコル処理工程及びデータ作成送出工程、またはデータ収集/送信処理工程及びデータ作成送出工程がオペレーティングシステム空間に実装されたものであることを特徴とする。

【0050】したがって、ユーザ空間にアプリケーションとして実装する場合に比べて優先的に実行されると共にメモリ等の使用頻度が非常に少なくなるので、処理速度を格段に向上させることができる。

【0051】上記目的を達成するために、請求項4記載の発明に係るネットワーク通信の監視・制御装置は、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析する装置であって、ネットワークに対してパケットを送受信できるデータ収集/送信処理部と、制御メッセージによって書き換えられるアナライザ管理情報を基にデータ収集/送信処理部との間でパケット通信を行うとともに、通信に不正が検知されたときにはパケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理部と、アナライザ管理部から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータをアナライザ管理部に伝送するプロトコル処理部と、プロトコル処理部から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理部に送出するアプリケーションアナライザとを備えたことを特徴とする。

【0052】したがって、パケットから解析データを再構築して解析できるので、再構築後のデータにより不正の判断を行うことができる。また、デュアルホームホストの採用によりハイジャック機能を実現できるので、ネットワークでの不正を防止することができる。

【0053】請求項5記載の発明は、請求項4において、データ収集/送信処理部からのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにしてデータ収集/送信処理部に与えるプロトコルスタックと、プロトコルスタックから解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データをプロトコルスタックに与えるネットワークアプリケーションとを備えたことを特徴とする。

【0054】請求項6記載の発明は、請求項4または5において、データ収集/送信処理部、アナライザ管理部、プロトコル処理部及びプロトコルスタック、ないし、データ収集/送信処理部及びプロトコルスタックがオペレーティングシステム管理空間に実装されたものであることを特徴とする。

【0055】したがって、ユーザ空間にアプリケーションとして実装する場合に比べて優先的に実行されると共にメモリ等の使用頻度が非常に少なくなるので、処理速度を格段に向上させることができる。

【0056】請求項7記載の発明に係るコンピュータに実行させるためのプログラムを記録したコンピュータ読み取り可能な記憶媒体は、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析する方法であって、ネットワークに対してパケットを送受信できるデータ収集/送信処理手順と、制御メッセージによって書き換えられるアナライザ管理情報を基にデータ収集/送信処理手順との間でパケット通信を行うとともに、通信に不正が検知されたときにはパケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理手順と、アナライザ管理手順から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータをアナライザ管理手順に伝送するプロトコル処理手順と、プロトコル処理手順から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理手順に送出するアナライザ手順とをコンピュータに実行させるためのプログラムを記録している。

【0057】したがって、コンピュータで記録媒体に格納されたプログラムを読み取り、インストールすることによって、あるいはそのままプログラムを実行することによって、ネットワーク伝送時にデータ部分に発生した不正を容易に判別することができる。

【0058】請求項8の発明は、請求項7において、データ収集/送信処理手順からのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにしてデータ収集/送信処理手順に与えるデータ作成送出手順と、データ作成送出手順から解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データをデータ作成送出手順に与えるネットワーク処理手順とをコンピュータに実行させるためのプログラムを記録した記憶媒体である。

【0059】

【発明の実施の形態】以下、本発明の構成を図面に示す実施の形態に基づいて詳細に説明する。

【0060】〔第1の実施の形態〕図2に、本発明に係るネットワーク不正解析方法が適用されたネットワークシステムの一例を示す。

【0061】この図2において、符号1は構内ネットワーク（以下、「LAN」という）であり、このLAN1にはルータ2を介して緩衝帯ネットワーク3が接続されている。この緩衝帯ネットワーク3は、ルータ6を介して外部ネットワーク7に接続されている。

【0062】本実施形態ではLAN1は、複数のステー

ション（図示せず）と、これらステーションを結ぶネットワーク（図示せず）と、外部に接続する外部接続回線11とからなる。この外部接続回線11はルータ2に接続されている。但し、LAN1としては上述のものに限られず、単一のステーションから成るものとする事もできる。

【0063】緩衝帯ネットワーク3は、本発明に係るネットワーク不正解析方法を実現するネットワーク不正解析システム50と、コモンバス51と、外部接続回線52、53とからなる。ネットワーク不正解析システム50はコモンバス51に接続されている。外部接続回線52はルータ2とコモンバス51とを接続している。外部接続回線53はコモンバス51とルータ6とを接続している。

【0064】また、ネットワーク不正解析システム50は従来のネットワーク不正解析システムと同様にコンピュータシステムにて構成される。このコンピュータシステムは、特に図示していないが、例えば各種の演算処理を実行する中央演算処理装置と、演算処理を実行する上で必要なプログラムやデータ等を記憶する主記憶装置と、入出力ポート等の各種インタフェースと、このネットワーク不正解析システムを実現するプログラムやその処理を実行する上で必要な各種データや定数等を記憶する例えばハードディスクドライブ装置と、データや必要な指令を入力する入力装置と、中央演算処理装置で処理した結果を出力する出力装置とから構成されるのが一般的である。

【0065】図3に、本発明に係るネットワーク不正解析方法の実施の一形態を示す。この図3において、ネットワーク不正解析システム50は、中央演算処理装置が主記憶装置に記憶されたプログラムを処理することにより実現されるものであって、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析するシステムである。すなわち、ネットワーク不正解析システム50は、例えば緩衝帯ネットワーク3で伝送されているパケットPTを取り込むデータ収集工程を実行するデータ収集部55と、データ収集部55からのパケットPTaを各階層化モジュールでフィルタリング及び再構築することにより解析データDTを作成するデータ作成工程を実行するデータ作成部56と、データ作成部56からの解析データDTに不正が発生しているか判定して解析結果CDCを出力するデータ解析工程を実行するデータ解析部57とから構成されている。

【0066】図1に、ネットワーク不正解析システム50の詳細構成を示す。この図1において、データ収集部55は、例えば緩衝帯ネットワーク3の間で伝送されているパケットPTを取り込むデータ収集処理部551と、このデータ収集処理部551のデータ収集処理を制御するデータ収集制御部552とから構成されている。

データ収集処理部551は、収集したパケットPTをデータ作成部56に供給する。

【0067】データ作成部56は、階層化されたプロトコルに応じた階層化モジュールのパラメータを、予め読み込んでおいたコンフィグレーションファイルで指定された情報に基づいて設定し、データ収集部55からのパケットPTaを各階層化モジュールでフィルタリングしてパケットPTaの細分化されたデータを元の単位に再構築することにより解析データDTを作成するデータ作成工程を実行するものである。

【0068】このデータ作成部56は、大別すると、階層化モジュール561と、これを制御し管理する制御管理部562とから構成されている。制御管理部562は、予め読み込んでおいたコンフィグレーションファイルで指定された情報に基づいて、階層化されたプロトコルに応じた階層化モジュール561のパラメータを設定できる。また、階層化モジュール561は、設定に基づいてデータ収集部55からのパケットPTaをフィルタリングしてパケットPTaの細分化されたデータを元の単位に再構築することにより解析データDTを作成できるようにになっている。

【0069】階層化モジュール561は、1レイヤモジュール61₁, ..., kレイヤモジュール61_k, ..., Nレイヤモジュール61_Nからなる複数のレイヤモジュールから構成されている。ここで、Nは任意の整数であり、kは1~Nの間の任意の整数である。

【0070】1レイヤモジュール61₁は、読み込まれたコンフィグレーションファイルの内容に応じて、1レイヤフィルタ61_{1A}および1レイヤ再構築部61_{1B}が構成される。kレイヤモジュール61_kは、読み込まれたコンフィグレーションファイルの内容に応じて、kレイヤフィルタ61_{kA}およびkレイヤ再構築部61_{kB}が構成される。Nレイヤモジュール61_Nは、読み込まれたコンフィグレーションファイルの内容に応じて、Nレイヤフィルタ61_{NA}およびNレイヤ再構築部61_{NB}が構成される。そして、各レイヤモジュール61₁, ..., 61_k, ..., 61_Nは、1からNまで順に処理される。

【0071】制御管理部562は、各レイヤモジュール61₁, ..., 61_k, ..., 61_Nにおいてデータを作成するデータ作成制御部62aと、各レイヤモジュール61₁, ..., 61_k, ..., 61_Nにおいてデータの記憶を制御するメモリ制御部62bと、解析データDTを出力させるためのアナライザ管理部62cとから構成されている。そして、制御管理部562は、1レイヤモジュール61₁, ..., kレイヤモジュール61_k, ..., Nレイヤモジュール61_Nをそれぞれ制御及び管理する。

【0072】データ解析部57は、予め読み込んでおいたコンフィグレーションファイルで指定された内容を基にデータ作成部56からの解析データDTに不正が発生しているか判定して解析結果CDCを出力するデータ解

析工程を実行するものである。

【0073】このデータ解析部57には、アプリケーション層セッション単位のデータに不正が有るかを解析するアプリケーションアナライザ570が設けられている。そして、このアプリケーションアナライザ570は、予め読み込んだコンフィグレーションファイルに記載されている内容に基づいてフィルタリング及び再構築されたデータDTに不正がないか否かを解析する。

【0074】図4に、コンフィグレーションファイルの例を示す。この図4において、コンフィグレーションファイル500は、データ作成部56からデータ解析部57に渡すデータDTの指定をするための情報やデータ解析部57で解析する情報を書き込むことができるようになっており、例えば図4に示すような内容を記載できるようになっている。

【0075】図5に、解放型システム間相互接続(OSI; Open System Interconnection)の7層モデルの例を示す。この図に示す階層化モデル800は、解放型システムの通信機能を7つに階層化したものであり、物理レイヤ801、データリンクレイヤ802、ネットワークレイヤ803、トランスポートレイヤ804、セッションレイヤ805、プレゼンテーションレイヤ806、アプリケーションレイヤ807に階層化されている。

【0076】物理レイヤ801は、ビット伝送に必要な物理的条件や電気的条件が定められている。データリンクレイヤ802は、データ伝送誤り制御手順等が定められている。ネットワークレイヤ803は、公衆バケット交換等各種通信網を介して両端のシステム間でデータのやりとりを可能とするためのものである。トランスポートレイヤ804は、両端のシステムの間で透過的に信頼性の高いデータ転送を行うためのものである。セッションレイヤ805は、両端のアプリケーションプロセスでの対話を効率よく行うため、同期をとったり伝送モードの選択、送信権の制御を行うためのものである。プレゼンテーションレイヤ806は、両端のアプリケーションプロセスが扱うデータを正確にかつ効率よく転送できるようにデータ形式を制御するためのものである。アプリケーションレイヤ807は、ユーザが実行する様々なアプリケーションに応じて、ファイル転送、メッセージ通信処理システム、下層端末、遠隔データベースアクセスなどのアプリケーションサービス要求を実行するためのものである。

【0077】このように構成されたネットワーク不正解析システム50の動作を図6～図8に説明する。図6に、同システム50の動作を説明するためのフローチャートを示す。図7、8に、同システムによって処理されるデータの内容の例を示す。

【0078】ここで、緩衝帯ネットワーク3のコモンバス51上を一つの意味有る情報ないしデータを構築する複数のデータバケット、例えば通信データ90a、90

b、90c、90dが流れているとする。

【0079】これらの通信データ90a、90b、90c、90dは、例えば図7に示すように受信先ID91a、91b、91c、91dと、送信元ID92a、92b、92c、92dと、順番93a、93b、93c、93dと、データ94a、94b、94c、94dとから構成されている。

【0080】このコモンバス51上の伝送される通信データ90aは、次のようにして構成されている。すなわち、例えば「I am a cracker.」というアプリケーションレイヤでのデータ95は細分化されて(ステップ(S)21)、「I am」というデータ94bと、「a」というデータ94dと、「rac」というデータ94aと、「ker.」というデータ94cとに分割されたデータとなる。そして、さらに細分化されて(S22)、ヘッダを付けた通信データ90a、90b、90c、90dにされ、これらは図5に示す階層化モデル800によりネットワーク上を伝送されることになる。緩衝帯ネットワーク3上でのデータ単位は、通信データ90a、90b、90c、90dとなる。このデータ単位で、緩衝帯ネットワーク3のコモンバス51上を伝送されることになる。

【0081】図6に示すように、ネットワーク不正解析システム50は、コンフィグレーションファイルを読み込む(S11)。これにより、データ作成部56の1レイヤモジュール61₁の1レイヤフィルタ61_{1a}及び1レイヤ再構築部61_{1b}、…、kレイヤモジュール61_kのkレイヤフィルタ61_{ka}及びkレイヤ再構築部61_{kb}、…、Nレイヤモジュール61_NのNレイヤフィルタ61_{Na}及びNレイヤ再構築部61_{Nb}には、データ解析部57に渡すデータやフィルタリング及び再構築すべきデータに関するパラメータが設定される。また、このコンフィグレーションファイルにより、データ解析部57で解析する内容が設定される。

【0082】ここで、Nの値は、図5に示す階層化モデル800のどの階層までのデータとして再構築するかにより決定される。本実施形態では、N=7としてアプリケーション層セッション単位にまで再構築している。但し、N=7に限られず、例えばN=3としてネットワーク層セッション単位にまで再構築するようにもできる。いずれの場合もデータ解析に必要なセッション単位にまで再構築するように設定できる。

【0083】次に、データ収集部55は、データ収集制御部552の制御下に、コモンバス51の上を伝送されているバケットPTの1バケット(例えば通信データ90a)を取り込み(S12)、この1バケットがコンピュータシステムの主記憶装置またはハードディスクドライブ装置等にある作業エリアにコピーされる(S13)。これにより、この作業エリアにコピーされた1バケットは、階層化モジュール561の1レイヤモジュール

ル61_iに取り込まれ、コンフィグレーションファイルの情報に応じて1レイヤ目の処理を実行して次の層のレイヤモジュールに渡す(S14)。この1レイヤモジュール61_iでは、パケットのヘッダ部のさまざまなフィールドの中の値と、コンフィグレーションファイルで指定されたフィルタ通過パラメータとの比較を行い、コンフィグレーションファイルで指定されているデータを選びだす処理を実行したり、通過処理したりする。また、フィルタ通過処理ではデータを変更しない。

【0084】次いで、1レイヤモジュール61_iの下層の各レイヤモジュールは、上述と同様にコンフィグレーションファイルの内容に応じてそれぞれ処理を実行する。

【0085】kレイヤモジュール61_kの前のモジュールが処理した結果をkレイヤモジュール61_kが受け取る。

【0086】ここで、各レイヤモジュール61₁, ..., 61_k, ..., 61_nでは同様の処理が実行されるので、一例としてkレイヤモジュール61_kについて詳細に説明する。

【0087】kレイヤモジュール61_kは、予め読み込まれたコンフィグレーションファイルの情報に応じて、受け取ったデータをkレイヤフィルタ61_{ka}に通すのか否か、kレイヤ再構築部61_{kb}を動作させるのか否か、あるいは再構築あるいは非再構築のデータをデータ解析部57に渡すか否かのパラメータが設定されているので、そのパラメータの設定に従って処理を実行する(S15)。

【0088】例えば、コンフィグレーションファイルの情報の指定によって各パラメータが設定されているとすると、kレイヤモジュール61_kでは、上層のレイヤモジュールが処理した結果のデータをkレイヤフィルタ61_{ka}を通過させたのち(S151; YES)、そのデータを再構築する場合には(S152; YES)、バッファを使用して再構築し(S153)、アプリケーション層単位でのデータにまでの再構築が完了しない場合には(S154; NO)、kレイヤモジュール61_kを抜ける。そして、次のパケットを読み込む(S12)。

【0089】また、バッファを使用して再構築(S153)、再構築が完了した場合(S154; YES)にはkレイヤ再構築部61_{kb}のバッファにデータが蓄えられている。この場合あるいはデータを再構築しない場合(S152; NO)には、再構築したデータをデータ解析部57に渡すタイミグになったときに(S155; YES)、kレイヤ再構築部61_{kb}のバッファに蓄えていたデータをデータ解析部57に渡す(S156)。また、データ解析部57に渡さないタイミグになったところで(S155; NO)、これより上側の層のレイヤモジュールにデータを渡す。

【0090】これにより、kレイヤモジュール61_kよ

り上層のレイヤモジュールで処理された結果、図8に示すように再構築されて(S23)、分割されたデータとして「rac」というデータ94aがデータ解析部57に渡る。

【0091】以上の処理を1パケット毎に繰り返すことにより(S13~S16)、ネットワーク上でのデータ単位のパケットPTは、図8に示すように再構築され(S23)、「I am」というデータ94bと、「ac」というデータ94dと、「rac」というデータ94aと、「ker.」というデータ94cとに分割されたデータとなる。さらに、再構築されて(S24)、「I am a cracker.」というアプリケーションレイヤでの元のデータ95にされる。

【0092】下位の層のレイヤモジュールが上記のようにコンフィグレーションファイルの情報によって各パラメータが設定されているとすると、Nレイヤフィルタ61_nにまでデータが渡る。

【0093】Nレイヤフィルタ61_nでは、上述と同様にコンフィグレーションファイルの情報により設定されたパラメータで受け取ったデータの処理を行い、解析データDTとしてデータ解析部57に渡す(S16)。

【0094】また、このフローチャートを使用すれば、複数の種類のフィルタを設定することにより複数の種類のクラッキング等の不正アクセスを同時に監視できる。

【0095】データ解析部57では、アプリケーションアナライザ570が予め読み込んでおいたコンフィグレーションファイルで指定された内容を基にデータ作成部56からの解析データDTに不正が発生しているか判定している。このアプリケーションアナライザ570の判定した結果は、例えばディスプレイ上に、図9に示すような画面900として表示される。

【0096】ここで、アプリケーションアナライザ570は、不正解析処理を次のいずれか一方あるいは双方を採用するようにしている。但し、これらの処理方法に限られないのは勿論である。

【0097】(i)注目しているプロトコルでどのような操作をすれば、不正が実行可能かを調べておき、その手順をアプリケーションアナライザ570用のコンフィグレーションファイルに記述しておき、解析データDTとコンフィグレーションファイルに記載された手順を比較して一致すれば、その通信は不正なものであると判断処理を実行する。

【0098】(ii)セキュリティを確保するために厳重に管理する必要があるファイルやプログラム等をアプリケーションアナライザ570用のコンフィグレーションファイルに記載しておき、記載しておいたリソースに変化があった時点で、その通信は不正なものであると判断処理を実行する。

【0099】上述したネットワーク不正解析システム50によれば、例えば次のような不正(クラッキング)を

防止できる。

【0100】(1) TCPプロトコルのシーケンス番号を推測し、偽のセッションを確保し攻撃対象ホストのファイルに不正アクセスすること。

【0101】(2) SMTPプロトコルのDEBUGコマンドを使い、攻撃対象ホストのファイルに不正にアクセスすること。

【0102】(3) MIMEを使って攻撃ホストのセキュリティ管理ファイルを書換え、攻撃ホストにパスワード無しでログインできるようにすること。

【0103】(4) 電子メールやニュースを攻撃対象ホストの処理能力を超えて大量に送りつけ、ネットワーク機能を麻痺させること。

【0104】(5) telnetプロトコルで、root, guest等めぼしいアカウントにアクセスし、パスワードを適当に推測し、ログインを試みること。

【0105】(6) TFTPプロトコルを用いて、パスワード無しで攻撃対象ホストのファイルにアクセスすること。

【0106】以上のように構成された本実施形態のネットワーク不正解析システム50によれば、次のような利点がある。

【0107】(1) アプリケーション層までのセッションの再構築を行うことができ、不正解析をおこなう者にとってわかり易いデータを取り扱えるようになった。すなわち、ネットワーク上でのパケットでのデータは細分化されているため解析を行うことが困難であるが、本実施形態のネットワーク不正解析システム50ではアプリケーション層までのセッションの再構築を行っているので解析データを意味有るデータとすることができる。これにより、データ中の不正の有無を容易に判定することができる。しかも、ネットワークでの不正の多くはデータ部分に発生するので、このデータ部分の不正を検出することにより効率の良い不正解析を行うことができる。

【0108】(2) 層毎の完全なモジュール化により、複数の種類のプロトコルを扱えるようになった。すなわち、ネットワークのプロトコルは図5に示す階層化モデル800のようにモデル化でき、本実施形態のネットワーク不正解析システム50はコンフィグレーションファイルの設定に基づきこのモデルを利用して不正の解析を実行するので、新たなプロトコルが開発されても、コンフィグレーションファイル及びデータ作成部56の新たなプロトコルに対応する層を担当する一部のレイヤモジュールのみを変更することにより対応することができる。このため、データ作成部56を新たに開発されたプロトコルに対応させるためにデータ作成部56の全体を変更する必要はないので、プロトコル数の増大に容易に対応することができる。

【0109】(3) データ解析部での不正解析はコンフィグレーションファイルに記述した検知方法に基づいて

行われるので、新たな種類の不正の発生をコンフィグレーションファイルの記述を変更するのみでデータ解析部のソフトウェア自体を変更することなく検出することができる。

【0110】(4) 緩衝帯ネットワーク3との組み合わせにより、任意の通信を取り扱えるようになった。例えば、LAN1のステーションや外部ネットワーク7が複数ある場合に、任意の送信元と受信先とについて不正解析を行うことができる。

【0111】(5) 緩衝帯ネットワーク3としてホストから独立しているので、ホスト数の増加にも容易に対応できる。

【0112】以上のように本発明の第1の実施の形態によれば、ネットワークの不正解析が可能になる。

【0113】〔第2の実施の形態〕図10に、本発明に係るネットワーク不正解析方法の第2の実施の形態を示す。この図10において、ネットワーク不正解析システム50は、中央演算処理装置が主記憶装置に記憶されたプログラムを処理することにより実現されるものであって、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワークにおける不正を解析するシステムである。なお、図10中で、実線に矢印は「データの流れ」を、破線に矢印は「制御メッセージの流れ」をそれぞれ示している(図30及び図34～図37でも同様)。

【0114】すなわち、ネットワーク不正解析システム50は、データ収集/送信処理部V11と、アナライザ管理部V12と、プロトコル処理部V13と、アプリケーションアナライザV21と、プロトコルスタックV101と、ネットワークアプリケーションV201とから構成されている。ここで、データ収集/送信処理部V11と、アナライザ管理部V12と、プロトコル処理部V13と、プロトコルスタックV101とは、OS空間V1において実装されている。また、アプリケーションアナライザV21と、ネットワークアプリケーションV201とは、ユーザ空間V2に実装されている。

【0115】また、データ収集/送信処理部V11は、例えば緩衝帯ネットワーク3で伝送されているパケットPTを取り込む収集処理を実行し、あるいはアナライザ管理部V12からのパケットPTA2を整形してパケットPTA1を作成してネットワークに送信するデータ収集/送信工程を実行する。データ収集/送信処理部V11及びデータ収集/送信工程は、それぞれ第1の実施形態でのデータ収集部55及びデータ収集工程に相当する。

【0116】アナライザ管理部V12は、データ収集/送信処理部V11からのパケットPTについて各アプリケーションアナライザ毎にフィルタ処理やフォワード処理を行いパケットPTaを作成し、かつプロトコル処理部V13からの送信データDTAを整形してPTA2を

作成し、さらにプロトコル処理部V13からの制御メッセージCMを受け取りアナライザ管理情報を更新するアナライザ管理工程を実行する。

【0117】プロトコル処理部V13は、アナライザ管理部V12からのバケットPTaをアプリケーションアナライザの指定するプロトコル処理モジュールで再構築処理をすることにより解析データDTを作成するプロトコル処理工程を実行し、また、アプリケーションアナライザV21からの送信データDTA及び制御メッセージCMを受信して送出する処理を実行する。

【0118】これらアナライザ管理部V13及びプロトコル処理部V13は第1の実施形態でのデータ作成部56に相当すると共に、アナライザ管理工程及びプロトコル処理工程は第1の実施形態でのデータ作成工程に相当する。

【0119】アプリケーションアナライザV21は、プロトコル処理部V13からの解析データDTに不正が発生しているか判定して解析結果CDCを出力した解析結果からデータDTA及び制御メッセージCMを作成するデータ解析工程を実行する。アプリケーションアナライザV21及びデータ解析工程は、それぞれ第1の実施形態でのデータ解析部57及びデータ解析工程に相当する。

【0120】プロトコルスタックV101は、データ収集/送信処理部V11からのバケットPTbを取り込んで処理することにより解析データDTbを作成すると共に、ネットワークアプリケーションV201から送出データDTBを受け取って処理して送出バケットPTBを作成してデータ収集/送信処理部V11に与えるデータ作成送出工程を実行する。

【0121】ネットワークアプリケーションV201は、プロトコルスタックV101からの解析データDTbを取り込み所定の処理を実行して送出データを作成すると共に、送出データDTBをプロトコルスタックV101に出力するネットワーク処理工程を実行する。

【0122】このようなネットワーク不正解析システム50は、図10に示すように、通常のネットワークアプリケーションによる通信について、データ収集/送信処理工程と、アナライザ管理工程と、プロトコル処理工程と、アナライザ工程とを実行すると共に、データ作成送出工程及びネットワーク処理工程を実行する。このようなネットワーク不正解析システム50を構成する各要素により、第1の実施形態と同様の手法も用いて通信内容を解析した結果、不正行為が行われているセッションやホスト等を特定することができる。

【0123】図11及び図12に、上記ネットワーク不正解析システム50の詳細構成を示す。この図11において、実線に黒塗矢印は「データの流れ」を、破線に矢印は「制御メッセージの流れ」を、実線に細い矢印は「情報の参照/変更」をそれぞれ示している。

【0124】これらの図において、アナライザ管理部V12は、データ送信処理部V12aと、メッセージ処理部V12bと、フィルタ処理部V12cと、フォワード処理部V12dと、アナライザ管理情報V12eとからなる。また、アナライザ管理情報V12eには、アプリケーションアナライザA用の管理情報V12ea、アプリケーションアナライザB用の管理情報V12eb、及びアプリケーションアナライザC用の管理情報V12ecを備えている。

【0125】上記データ送信処理部V12aは、例えばデータ収集/送信処理部V11からのバケットPTを取り込み各アプリケーションアナライザ毎にフィルタ処理を行う。メッセージ処理部V12bは、フィルタ処理部V12cからのバケットPTを取り込み各アプリケーションアナライザ毎にフォワード処理を行う。アナライザ管理情報V12eは、フィルタ処理部V12c及びフォワード処理部V12dから参照されまたメッセージ処理部V12bから参照及び更新される。メッセージ処理部V12bは、プロトコル処理部V13からのデータDTAを取り込みまた制御メッセージCMを取り込みアナライザ管理情報V12eを更新する。データ送信処理部V12aは、フォワード処理部からのバケットPTf及びメッセージ処理部からのバケットPTAを取り込み整形を行いバケットPTA2を出力する。また、アナライザ管理情報V12eは、例えばアプリケーションアナライザA(V21a)用の管理情報V12ea及びアプリケーションアナライザB(V21b)用の管理情報V12eb及びアプリケーションアナライザC(V21c)用の管理情報V12ecとから構成されている。

【0126】プロトコル処理部V13は、プロトコル処理モジュールV13aと、プロトコル処理モジュールV13c1と、プロトコル処理モジュールV13c2とからなる。

【0127】プロトコル処理モジュールV13aは、例えば、アプリケーションアナライザ(A)V21aによって指定される。プロトコル処理モジュールV13c1及びプロトコル処理モジュールV13c2は、アプリケーションアナライザ(C)V21cによって指定される。なお、アプリケーションアナライザ(B)V21bのように、指定するプロトコル処理モジュールが存在しない場合もある。

【0128】アプリケーションアナライザV21は、例えば、アプリケーションアナライザ(A)V21aと、アプリケーションアナライザ(B)V21bと、アプリケーションアナライザ(C)V21cとから構成されている。これらは、既に述べたが、ユーザ空間V2に存在する。

【0129】図13に、アナライザ管理情報の例を示す。この図13において、アナライザ情報格納用構造体ST1は、フィルタ処理部V12cからプロトコル処理

部V13に渡すバケットPTaの指定をするための情報や、フォワード処理部V12dからデータ送信部V12aへ渡すバケットPTfの指定をするための情報を書き込むことができるようになっており、例えば図13に示すような内容のように記載される。

【0130】このように構成されたネットワーク不正解析システム50の動作を図14～図17を参照して説明する。図14及び図15に、同システム50におけるネットワーク4からのバケットを処理する場合の動作を説明するためのフローチャートを示す。図16に、同システム50におけるアプリケーションアナライザV21からのデータを処理する場合の動作を説明するためのフローチャートを示す。図17に、図14～図16に現れるデータ送信部V12aの動作を説明するためのフローチャートを示す。

【0131】図14及び図15に示すように、ネットワーク不正解析システム50は、バケットを取り込む(P1)。取り込んだバケットについて、フィルタ処理部V12cにおいて、未処理のアナライザが存在する場合は(P11; Yes)、未処理のアナライザ情報を取得する(P12)。未処理のアナライザが存在しない場合は(P11; No)、フォワード処理部V12dへバケットを送信する。未処理のアナライザ情報を取得したのち、その情報に従ってバケットがアナライザのフィルタリングルールを満たすか否かを判定し、満たす場合は(P13; Yes)、バケットをプロトコル処理部へ送信する。満たさない場合は(P13; No)、次のアナライザに対する処理に移る。

【0132】プロトコル処理部V13は、フィルタ処理部V12cからのバケットを受け取り、アナライザに対応するプロトコル処理モジュールがアナライザが解析しやすいデータ形式にバケットを処理し(P21)、データをアナライザへ転送する(P22)。

【0133】フォワード処理部V12dは、フィルタ処理部V12cから受け取ったバケットについて、未処理の禁止則が存在する場合(P31; Yes)、未処理の禁止則を取得する(P32)。未処理の禁止則が存在しない場合は(P31; No)、バケットをデータ送信処理部V12aへ送る。未処理の禁止則を取得したのち、バケットが禁止則に合致するか否かを調べ、合致する場合(P33; Yes)は、データを破棄し(P34)、次のバケットを入力する。データが禁止則に合致しない場合(P33; No)は、次の未処理の禁止則が存在するか調べる。そして、全ての禁止則に合致しない場合は、データをフォワードする。

【0134】図16に示すように、メッセージ処理部V12は、アナライザが送信したメッセージを受信する(P2)。受信したメッセージの内容を処理した後(P41)、アナライザ管理情報に変更があった場合は(P42; Yes)、アナライザ管理情報の変更処理を行う

(P43)。情報変更がなかった場合(P42; No)、または、アナライザ管理情報の変更を完了した場合(P42; Yes)、次にデータ送信が必要な場合(P44; Yes)、データをデータ送信処理部V12aへ渡す。データ送信が不要な場合は(P44; No)、何もしない。

【0135】図17は、図14～図16に共通に現れるデータ送信処理部の動作を示したフローチャートである。図17に示すように、データ送信処理部はデータを受け取り(P51)、通信可能となるようにデータを整形し(P52)、実際にネットワークへデータを送る(P53)。

【0136】また、このフローチャートを使用すれば、複数の種類のアプリケーションアナライザを設定することにより、複数の種類のクラッキング等の不正アクセスを同時に監視できる。

【0137】図18に、上述したネットワーク不正解析システム50を適用した通信装置MCNCを、ネットワークNTWで接続された通信装置MCNA、MCNBに適用した例を示す。

【0138】上述した実施の形態を備えた通信装置MCNCは、OS空間にCAプロトコルスタックを実装し、CA通信ライブラリ及びアナライザをユーザ空間に実装したため、この発明をユーザが簡単に利用できるユーザインタフェースを提供することができる。ここで、CAプロトコルスタックとは、図10に示すアナライザ管理部V12及びプロトコル処理部V13、または図34に示すUアナライザ管理部V31及びUプロトコル処理部V32、あるいは図36に示すプロトコルスタック*V41を意味する。

【0139】図22に、UNIXのソケットライブラリの場合を例として、疑似コードにより、通信の解析を行う場合のプログラミングの実際を示す。

【0140】同図(a)に示すように、本発明の通信ライブラリでは、socket()関数の呼び出しにより、アプリケーションアナライザはCAプロトコルスタックと接続する。これは初期状態である。

【0141】そして、bind()関数の呼び出しにより、フィルタルールを登録してデータ収集状態に移行する。さらに、accept()関数は、フィルタルールに合うセッションが開始されるまでスリープする。目的のセッションが開始されると、accept()関数の呼び出しから戻り、新たな識別子new_fdを得る。このnew_fdを使ってセッションのデータを読み込むことができる。

【0142】ところが、new_fdを使い特定セッションの解析を行うと、新たなフィルタルールに合うセッションが現われたときに対応することができない。そこで、特定セッションの解析は別のプロセスで行うことになる。それがfork()コールである。fork()によりプロセスは親と子に別れる。子の場合、fork()の戻り値は0である。

親の場合は0ではない。子はnew_fdを使って特定セッションを解析して親は再びaccept()をコールしフィルタールに合うセッションを待つ。

【0143】一方、図22(b)に示す従来の通信ライブラリでは、socket()関数の呼び出しにより、ネットワークアプリケーションはプロトコルスタックと接続する。そして、bind()関数の呼び出しによりサービスを指定する。さらに、accept()関数を呼び出して、クライアントからの接続を待つ。クライアントからの接続要求があるとaccept()関数の呼び出しから戻り、新たな識別子new_fdを得る。このnew_fdを使ってクライアントと通信することができる。

【0144】ところが、new_fd を使い特定クライアントと通信すると、新たなクライアントからの接続要求に答えることができない。そこで、特定クライアントとの通信は別のプロセスで行うことになる。そのため、fork()を使う。fork()によりプロセスは親と子に別れる。子の場合、fork()の戻り値は0である。親の場合は0ではない。子はnew_fdを使って特定クライアントと通信し、親は再びaccept()をコールしフィルタールに合うセッションを待つ。

【0145】上述のように従来のネットワーク解析ツールでは独自のコードスタイルにならざるを得なかったが、本発明のプロトコルスタックと通信ライブラリによりネットワークアプリケーションと同様のスタイルのプログラミングでネットワーク解析ツールを作成することができるようになる。

【0146】図23にUNIXのソケットライブラリの場合を例として、疑似コードにより、通信のハイジャックを行う場合のプログラミングの実際を示す。

【0147】同図に示すように、hijack()関数の呼び出しによりハイジャック状態に移行する。この状態ではnew_fdを使いデータを送信することができる(write()コール)。もちろん、データ収集状態の時に行えたデータ読み込みも継続して行える。write()コールの引き数に送信先というものがあるが、これでサーバまたはクライアントに対しての送信なのかを指定する。

【0148】従来はハイジャック機能の実現のために高度なプログラミング技術が必要で、また動作の安定するものは困難であった。本発明により、ネットワーク通信のハイジャックを行うツールをネットワークアプリケーションプログラムと同様のスタイルで、容易に開発することができるようになる。

【0149】さらに、上述したネットワーク不正解析システム50のアプリケーションアナライザの出力例を図24～図26に示す。

【0150】図24及び図25に示す

```
ETHER:00:808:17:88:8d >> 00:a0:24:23:16:0d
IP:202.26.75.116 >> 202.26.75.71
TCP:40508 >> 23
```

の部分はホストA (IPアドレスは202.26.75.116) からホストB (IPアドレスは202.26.75.71) へ、telnet (サービスポート番号23) で接続したことを意味する。ここでは、クライアントがホストAで、サーバがホストBである。また、SRC(source)で表わされるものがホストAで、DST (destination) で表わされるものがホストBである。

【0151】[src << (DST)]の部分、DSTからSRCへデータが渡ってきたことを意味する。SRCからDSTへのデータは[(SRC) >> dst]と成る。そして、(id:4553)はこの通信の識別子である。<len:71>は、送られてきたデータの長さを表す。上記の場合は71バイトのデータを表す。

【0152】これらの図は、ネットワークのデータを読み込み表示するツールの一例である。ホストAがホストBにtelnetで接続し、ホストBが

BSDI BSD/OS 3.0 (netlog.nal.go.jp)(tty0)

login:

をホストAに送る。ホストAからrot という入力があり、ホストBはPassword:というデータをホストAに返す、そこで、ホストAからroot1という入力があり、ホストBはlogin incorrectを返している。

【0153】次に、図26の表示内容について説明する。左のウィンドウは解析対象を指定するウィンドウである。EthernetアドレスやIPアドレス、TCPポート番号を入力できるフィールドがある。同図の例では、IPアドレス202.26.75.71の25番ポート宛の通信すべてを解析対象にしている。そして、左のウィンドウの下半分に解析対象に合致したセッションが表示される。同図の例では、202.26.75.116から202.26.75.71までのセッションが検出されている。検出されているセッション一覧のなかで、指定したものの詳細を別ウィンドウに表示することができる。複数のセッションが存在する場合は、ウィンドウをそれぞれに対応させて、複数開くことができる。

【0154】また、このウィンドウが図の右側に表示されている。このウィンドウの上部に通信に使用されているアドレスが表示され、中央部に通信の内容がリアルタイムで表示されている。最下部に2つのボタンがあり、[Kill Session]ボタンはセッションの強制終了を行い、[Hijack Session]はセッションのハイジャックを行う。この[Hijack Session]ボタンを押した後は、そのボタンの上のCmd to SrcやCmd to Dstフィールドにデータを入力し、それを通信の当事者に送ることができる。ハイジャック後も通信の当事者から出されたデータはウィンドウの中央部分に表示される。

【0155】〔第3の実施の形態〕図27に、同第3の実施の形態を示す。この図23では、ネットワーク不正解析システム50に複数のネットワークインターフェースを設けてマルチホーム化することができる。例えば2

つのネットワークインターフェースを設けてデュアルホーム化して別個のネットワーク4、5に接続されるときは、LAN1と外部ネットワーク7との間の通信は必ずネットワーク不正解析システム50の内部を通過することになる。このため、ネットワーク不正解析システム50でのバケットの収集を取りこぼし無く行うことができる。これにより、ネットワークでの不正を高精度に検出することができる。

【0156】また、図28に示すように、ネットワーク不正解析システム50をマルチホーム化すると共に、データ解析部57からデータ収集部55へ解析結果をフィードバックさせるようにしても良い。なお、図28では、データ作成部56及びデータ解析部57の具体的な図示は省略している。この場合、データ解析部57の解析結果によってデータ収集部55の処理が変化される。これにより、解析の結果、不正行為が行われているセッションやホスト等が特定できた場合、その通信についてはネットワーク不正解析システム50内を通過させないようにデータ収集部55を制御することができる。したがって、ネットワークでの不正のLAN1への侵入を防止することができる。

【0157】さらに、図29に示すように、データ作成部56を、層構造のレイヤフィルタ61_{1A}, ..., 61_{1A}, ..., 61_{1A}を有するフィルタリング処理部58と、層構造のレイヤ再構築部61_{1B}, ..., 61_{1B}, ..., 61_{1B}を有する再構築処理部59とを備えたものにする。この場合、データ作成部56では、フィルタリング処理部58においてフィルタリング処理をレイヤの数だけ先にまとめて行い、その後、再構築処理部59において各レイヤについての再構築処理を行う。このため、解析の対象にならないデータがより早く破棄されるため、不必要な再構築処理やメモリ占有が解消され、ネットワーク不正解析システム50の動作の高速化を図ることができる。

【0158】また、上述した第1の実施形態及び第3の実施形態ではネットワーク不正解析システム50は図示しないコンピュータシステムから構成されるものとしているが、これには限られない。例えば、上述したネットワーク不正解析方法のデータ収集工程とデータ作成工程とデータ解析工程とをコンピュータに実行させるためのプログラムを記録したCD-ROMやフロッピーディスク等のコンピュータ読み取り可能な記録媒体を使用して汎用コンピュータを作動させることによりネットワーク不正解析システムを実行することもできる。

【0159】ここでのプログラムは、具体的には、情報通信ステーションの間で階層化されたプロトコルにより通信を行うように構築されたネットワーク上で伝送されているバケットを取り込むデータ収集工程を実行するデータ収集手順と、階層化されたプロトコルに応じた階層化モジュールのパラメータを予め読み込んでおいたコン

フィグレーションファイルで指定された情報に基づいて設定し、データ収集手順からのバケットを各階層化モジュールでフィルタリングしてバケットの細分化されたデータを元の単位に再構築することにより解析データを作成するデータ作成工程を実行するデータ作成手順と、予め読み込んでおいたコンフィグレーションファイルで指定された内容を基にデータ作成手順からの解析データに不正が発生しているか判定するデータ解析工程を実行するデータ解析手順とをコンピュータに実行させるためのものである。

【0160】また、上述した実施形態ではデータ収集工程とデータ作成工程とデータ解析工程とを汎用コンピュータに実行させているが、これには限られずこれらの工程を専用のコンピュータから成る独立した装置に実行させるようにしても良い。さらに、上述した実施形態では、これらデータ収集工程とデータ作成工程とデータ解析工程とをコンピュータに実行させているが、これには限られずこれらの工程をシーケンス回路に実行させるようにしても良い。いずれの場合も、データ作成工程においてデータ収集工程からのバケットを各階層化モジュールでフィルタリングしてバケットの細分化されたデータを元の単位に再構築することにより解析データを作成するので、データ解析工程ではデータとして意味のある再構築後の解析データを解析対象とすることができる。このため、データ部分に発生した不正を容易に判別することができる。

【0161】〔第4の実施の形態〕図30に、同第4の実施の形態を示す。この図30に示す通信システムは、ネットワーク4、5の間に図10に示すネットワーク不正解析システム50を適用したものである。すなわち、ネットワーク不正解析システム50は、データ収集/送信処理部V11に、ネットワーク4と他のネットワーク5とを接続し、ネットワーク4とデータ収集/送信処理部V11との間で通信をし、ネットワーク5とデータ収集/送信処理部V11との間で通信をするようにデュアルホーム化したものである。

【0162】この図30に示すシステムでは、通信装置MCNAと通信装置MCNBとの間の通信は必ずネットワーク不正解析システム50を備えた通信装置MCNCを通過することになる。例えば、図31に示すように、通信装置MCNAから通信内容[Hello B!]、通信相手方[A->B]なるバケットが送出されてくると、通信装置MCNCは、そのバケットをCAプロトコルスタックに取り込み、CA通信ライブラリを介してアナライザに通信内容[Hello B!]を伝える。ここで、不正がない場合には、同一内容[Hello B!]に通信相手方[A->B]なるバケットを通信装置MCNBに伝送する。

【0163】次に、通信に不正があると判断したときには、通信装置MCNCはハイジャック状態になる。このとき、図32に示すように、通信装置MCNAから通信

内容[Hi B!]、通信相手方[A→B]なるパケットが送られてきて、また、通信装置MCNBから通信内容[Hello A!]、通信相手方[B→A]なるパケットが送られてきたとすると、これらはアナライザに取り込まれる。

【0164】図33に示すように、アナライザはこれまでの通信内容に基づいて、通信装置MCNAに送信すべき通信内容[0h A!]と通信装置MCNBに送信すべき通信内容[Bye B!]を作成する。そして、これら内容をアナライザからCAプロトコルスタックに渡す。プロトコルスタックは、通信内容[0h A!]は通信相手方[B→A]を付加して送出し、通信内容を[Bye B!]は通信相手方[A→B]を付加して送出する。

【0165】このようにしているため、通信内容を確実に収集でき、かつ、ハイジャックも確実に実行することができる。

【0166】ここで、図19及び図20に、この実施の形態が実現するハイジャック機能をネットワーク制御装置に適用した例を示す。図21に、ネットワーク制御装置の動作例を示す。

【0167】この図19(a)において、通信当事者TCNA、TCNBとの間にネットワーク制御装置NWCTが介在している通信システムにおいて、ネットワーク制御装置NWCTにアプリケーションアナライザV21から制御メッセージCMとしてフィルタルータ登録がなされる。このとき、ネットワーク制御装置NWCTに保持されるアプリケーションアナライザの状態は、図21に示すように、初期状態STS α からデータ収集状態STS β に移行する。

【0168】すると、アナライザ管理部V12は、図19(b)に示すように、ネットワーク制御装置NWCTを介して通信当事者TCNA、TCNBのデータを収集することになる。ネットワーク制御装置NWCTは、図21に示すように、データ収集状態STS β にある。

【0169】ここで、アナライザ管理部V12が通信に不正を検出したときに、その程度が小さいものについては、図20(a)に示すようにネットワーク制御装置NWCTを中断制御して転送中断にする。このとき、ネットワーク制御装置NWCTは、図21に示すように、データ収集状態STS β から転送中断状態STS γ に移行する。

【0170】ネットワーク制御装置NWCTは、転送中断状態STS γ において、収集したデータを上述した方法で解析し、不正がなければ、転送中断状態STS γ からデータ収集状態STS β に移行する。これにより、システムは、再び、図19(b)の状態に移行する。

【0171】一方、ネットワーク制御装置NWCTは、転送中断状態STS γ において、収集したデータを上述した方法で解析し、不正があると判断したときには、図21に示すように、ハイジャック状態STS δ に移行させる。この結果、図20(b)に示すように、通

信当事者TCNAはネットワーク制御装置NWCTを介してアプリケーションアナライザV21と通信し、また、通信当事者TCNBはアナライザ管理部V12と通信することになる。しかしながら、通信当事者TCNA及び通信当事者TCNB同士は互いの通信を行っているものと錯覚することになる。

【0172】なお、アナライザ管理部V12がデータ収集状態STS β にあるときに、明らかに不正がある通信であると判断したものについては、図21に示すように、データ収集状態STS β からハイジャック状態STS δ に直ちに移行させる。これにより、通信当事者TCNA、TCNBは、図19(b)の状態から図20

(b)の状態に移行することになる。

【0173】このように、通信当事者TCNA、TCNBの通信をアナライザ管理部V12で監視して、通信に不正を発見したときには、直ちにハイジャック状態にすることにより、不正な通信が行われることを防止することができる。

【0174】換言すると、このようなネットワーク不正解析システム50を構成する各要素により、通信内容を解析した結果、不正行為が行われているセッションやホスト等を特定することができた場合には、その通信については当該システム50内を通過させないようにしたりまた通信のハイジャックを行うようにアナライザ管理部V12を制御することができる。したがって、ネットワークでの不正のLAN1への侵入を防止することができる。

【0175】なお、上述の実施形態は本発明の好適な実施の一例ではあるがこれに限定されるものではなく本発明の要旨を逸脱しない範囲において種々変形実施可能である。

【0176】例えば、ネットワーク3に接続されたネットワーク不正解析システム50については、図34に示すように、Uアナライザ管理部V31とUプロトコル処理部V32をユーザ空間V2で実装するようにしてもよい。この場合、データ収集/送信処理部V11、プロトコルスタックV101については、図10の構成と同様にOS空間V1に実装しておくものとする。

【0177】また、ネットワーク4、5に接続されたデュアルホーム化したネットワーク不正解析システム50については、図35に示すように、Uアナライザ管理部V31とUプロトコル処理部V32をユーザ空間V2で実装するようにしてもよい。この場合、データ収集/送信処理部V11、プロトコルスタックV101については、図30の構成と同様にOS空間V1に実装しておくものとする。

【0178】あるいは、図34及び図35の実施形態で、データ収集/送信処理部V11やプロトコルスタックV101についてもユーザ空間V2で実装するようにしてもよい。

【0179】また、ネットワーク3に接続されたネットワーク不正解析システム50について、図36に示すように、既存のプロトコルスタックを改良してプロトコルスタック*V41にし、このプロトコルスタック*V41をOS空間V1に実装することにより、同様の作用効果を実現することができる。

【0180】さらにまた、ネットワーク4、5に接続されたネットワーク不正解析システム50について、図37に示すように、既存のプロトコルスタックを改良してプロトコルスタック*V41にし、このプロトコルスタック*V41をOS空間V1に実装することにより、同様の作用効果を実現することができる。

【0181】なお、アナライザ管理部やプロトコル管理部のみならずアプリケーションアナライザをもOS空間V1に実装することにより、同様の作用効果を実現することができる。

【0182】また、上述した第2の実施形態及び第4の実施形態ではネットワーク不正解析システム50は図示しないコンピュータシステムから構成されるものとしているが、これには限られない。例えば、上述したネットワーク不正解析方法のデータ収集/送信処理工程と、アナライザ管理工程と、プロトコル処理工程と、アナライザ工程とをコンピュータに実行させるためのプログラムを記録したCD-ROMやフロッピーディスク等のコンピュータ読み取り可能な記録媒体を使用して汎用コンピュータを作動させることによりネットワーク不正解析システムを実行することもできる。

【0183】ここでのプログラムは、具体的には、ネットワークに対してパケットを送受信できるデータ収集/送信処理手順と、制御メッセージによって書き換えられるアナライザ管理情報を基にデータ収集/送信処理工程との間でパケット通信を行うとともに、通信に不正が検知されたときにはパケットでもって通信当事者と個々に通信して通信当事者間の通信をハイジャックするハイジャック機能を備えたアナライザ管理手順と、アナライザ管理手順から得たパケットから解析データを再構築するとともに、制御メッセージ及びデータをアナライザ管理手順に伝送するプロトコル処理手順と、プロトコル処理手順から取り込んだ解析データに不正がないか判断するとともに、その判断結果に応じて制御メッセージ及びデータを作成してプロトコル処理手順に送出するアナライザ手順とをコンピュータに実行させるためのものである。

【0184】同様に、上述の各工程に加えてデータ作成送出工程とネットワーク処理工程をもコンピュータに実行させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体を使用して汎用コンピュータを作動させることによりネットワーク不正解析システムを実行することもできる。ここでのプログラムは、具体的には、上述の各手順に加えてデータ収集/送信処理手順か

らのパケットを取込み所定の処理をして解析データとし、送出データを受け取って送出パケットにしてデータ収集/送信処理手順に与えるデータ作成送出手順と、データ作成送出手順から解析データを受け取り所定の処理を実行するとともに、所定の処理をされた送出データをデータ作成送出手順に与えるネットワーク処理手順とをコンピュータに実行させるためのものである。

【0185】また、上述した各実施形態ではデータ収集/送信処理工程と、アナライザ管理工程と、プロトコル処理工程と、アナライザ工程との各工程、あるいはこれらに加えてデータ作成送出工程とネットワーク処理工程を汎用コンピュータに実行させているが、これには限られずこれらの工程を専用のコンピュータから成る独立した装置に実行させるようにしても良い。さらに、上述した実施形態では、これらの工程をコンピュータに実行させているが、これには限られずシーケンス回路に実行させるようにしても良い。いずれの場合も、ネットワークに発生した不正を容易に判別してハイジャック機能を実行することができる。

【0186】

【発明の効果】以上説明したように請求項1記載のネットワーク通信の監視・制御方法及び請求項4記載のネットワーク通信の監視・制御装置によれば、パケットから解析データを再構築して解析できるので、再構築後のデータにより不正の判断を行うことができる。また、デュアルホームホストの採用によりハイジャック機能を実現できるので、ネットワークでの不正を防止することができる。

【0187】また、請求項3記載のネットワーク通信の監視・制御方法及び請求項6記載のネットワーク通信の監視・制御装置によれば、ユーザ空間にアプリケーションとして実装する場合に比べて優先的に実行されると共にメモリ等の使用頻度が非常に少なくなるので、処理速度を格段に向上させることができる。

【0188】したがって、本発明のネットワーク通信の監視・制御方法によれば以下の効果を得ることができる。

(1) オペレーティングシステムのプロトコルスタックの拡張部分が用意する手段を用いることにより、任意の情報通信ステーション間の通信を解析でき、簡便なデータ解析が可能となる。すなわち、データの解析者自らが、ネットワーク上の細分化されたデータをセッション単位にまで再構築する必要がなく、オペレーティングシステムの機能呼び出すのみで任意の通信を任意のデータ形式で参照することが可能となる。

【0189】また、オペレーティングシステムの機能としてデータ参照機能の実現されているため、汎用的なインターフェースとなっており、簡便さや拡張性に富んだ実現方法となっている。

【0190】(2) 上記(1)の機能で発見された不正

通信のセッションについて、オペレーティングシステムの用意する手段により、通信のハイジャックが可能となっており、不正通信を簡便に防止できる。これは、オペレーティングシステムの機能と呼び出すのみで対象としている通信を二つの独立した通信分割し、それぞれの通信に対して、不正通信の詳細な解析を行うための通信を生成することが可能となる。

【0191】また、上述したように本発明に係るネットワーク通信の監視・制御装置によれば、次のような利点がある。

(1) オペレーティングシステムのプロトコルスタックの拡張部分が用意する手段を用いることにより、任意の情報通信ステーション間の通信を解析でき、簡便なデータ解析が可能となる。すなわち、データの解析者自らが、ネットワーク上の細分化されたデータをセッション単位にまで再構築する必要がなく、オペレーティングシステムの機能と呼び出すのみで任意の通信を任意のデータ形式で参照することが可能となる装置を提供できる。

【0192】(2) 不正通信に対しては、オペレーティングシステムの用意する手段により、通信のハイジャックが可能となっており、不正通信を簡便に防止できる装置を提供できる。

【0193】さらに、上記本発明に係るネットワーク通信の監視・制御プログラムを記録したコンピュータ読み取り可能な記録媒体によれば、コンピュータに読み込ませることにより、オペレーティングシステムのプロトコルスタックの拡張部分が用意する手段を用いることが可能となり、任意の情報通信ステーション間の通信を解析でき、簡便なデータ解析が可能となる。

【図面の簡単な説明】

【図1】本発明のネットワーク不正解析方法が適用されたネットワーク不正解析システムの実施の形態を示すブロック図である。

【図2】ネットワーク不正解析システムとネットワークとの接続を示すブロック図である。

【図3】ネットワーク不正解析システムの実施の形態の概略を示すブロック図である。

【図4】同実施の形態で使用するコンフィグレーションファイルの例を示す説明図である。

【図5】同実施の形態にプロトコルの階層化の例を説明するための図である。

【図6】同実施の形態の動作を説明するためのフローチャートである。

【図7】同実施の形態の動作を説明するための図である。

【図8】同実施の形態の動作を説明するための図である。

【図9】同実施の形態で処理した結果を示す図である。

【図10】同実施の形態におけるネットワーク不正解析システムの実施の形態の概略を示すブロック図である。

【図11】同実施の形態におけるネットワーク不正解析システムの実施の形態の詳細を示すブロック図である。

【図12】同実施の形態におけるネットワーク不正解析システムの実施の形態の詳細を示すブロック図である。

【図13】同実施の形態で使用する構造体の例を示す説明図である。

【図14】同実施の形態の動作を説明するためのフローチャートである。

【図15】同実施の形態の動作を説明するためのフローチャートである。

【図16】同実施の形態の動作を説明するためのフローチャートである。

【図17】同実施の形態の動作を説明するためのフローチャートである。

【図18】同実施の形態の動作を説明するための図である。

【図19】同実施の形態における通信ハイジャック機能の動作を説明するための図であり、(A)は初期状態、(B)はデータ収集状態を示す。

【図20】同実施の形態における通信ハイジャック機能の動作を説明するための図であり、(A)は転送中断状態、(B)はハイジャック状態を示す。

【図21】同実施の形態におけるネットワーク制御装置の動作状態遷移を説明するための図である。

【図22】同実施の形態の通信監視用のプログラミングスタイルの疑似コードを示す説明図であり、(A)は本発明の通信ライブラリについて、(B)は従来の通信ライブラリについて示す。

【図23】同実施の形態の通信ハイジャック用のプログラミングスタイルの疑似コードを示す説明図である。

【図24】同実施の形態で処理した結果を示す説明図である。

【図25】同実施の形態で処理した結果を示す説明図である。

【図26】同実施の形態で処理した他の結果を示す説明図である。

【図27】ネットワーク不正解析システムの他の実施形態を示すブロック図である。

【図28】ネットワーク不正解析システムの別の実施形態を示すブロック図である。

【図29】ネットワーク不正解析システムのさらに他の実施形態を示すブロック図である。

【図30】同実施の形態におけるネットワーク不正解析システムの実施の形態の概略を示すブロック図である。

【図31】同実施の形態における通信動作の例を示す説明図である。

【図32】同実施の形態における通信動作のうち通信ハイジャックの一態様例を示す説明図である。

【図33】同実施の形態における通信動作のうち通信ハイジャックの他の態様例を示す説明図である。

【図34】 同他の実施の形態におけるオペレーティングシステム空間の実装例を示すブロック図である。

【図35】 同他の実施の形態におけるオペレーティングシステム空間の他の実装例を示すブロック図である。

【図36】 同他の実施の形態におけるオペレーティングシステム空間のさらに他の実装例を示すブロック図である。

【図37】 同他の実施の形態におけるオペレーティングシステム空間のさらに他の実装例を示すブロック図である。

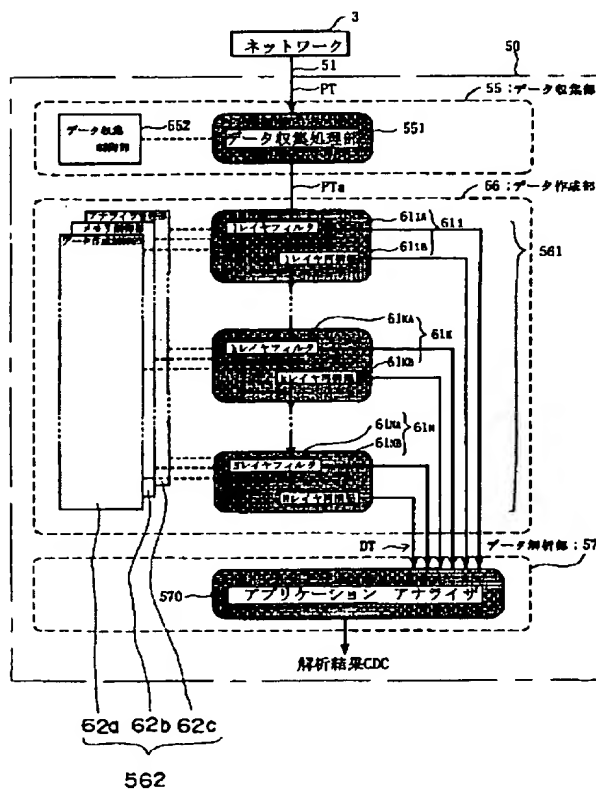
【図38】 従来のバケット単位の監視ツールの構成例を示すブロック図である。

【図39】 従来の代理サーバ形式の監視ツールの構成例を示すブロック図である。

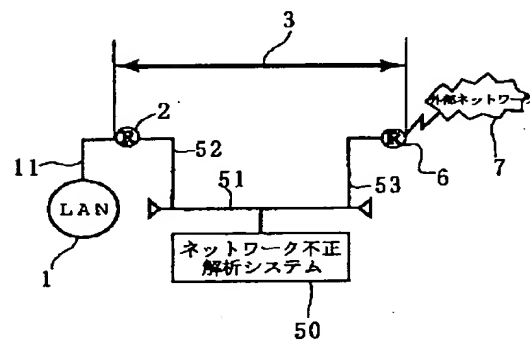
【符号の説明】

- 1 LAN
- 3 緩衝帯ネットワーク
- 7 外部ネットワーク
- 50 ネットワーク不正解析システム
- 55 データ収集部
- 56 データ作成部
- 57 データ解析部
- V1 オペレーティングシステム空間
- V2 ユーザ空間
- V11 データ収集/送信処理部
- V12 アナライザ管理部
- V13 プロトコル処理部
- V21 アプリケーションアナライザ
- V101 プロトコルスタック
- V201 ネットワークアプリケーション

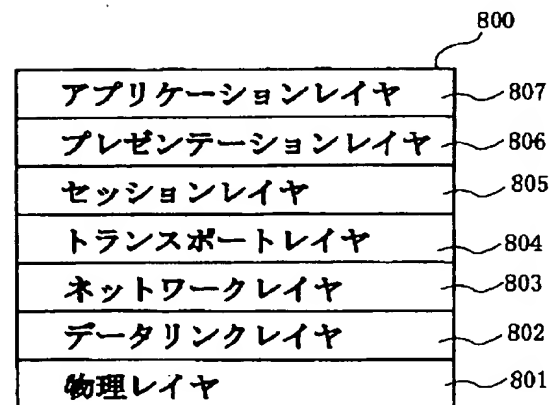
【図1】



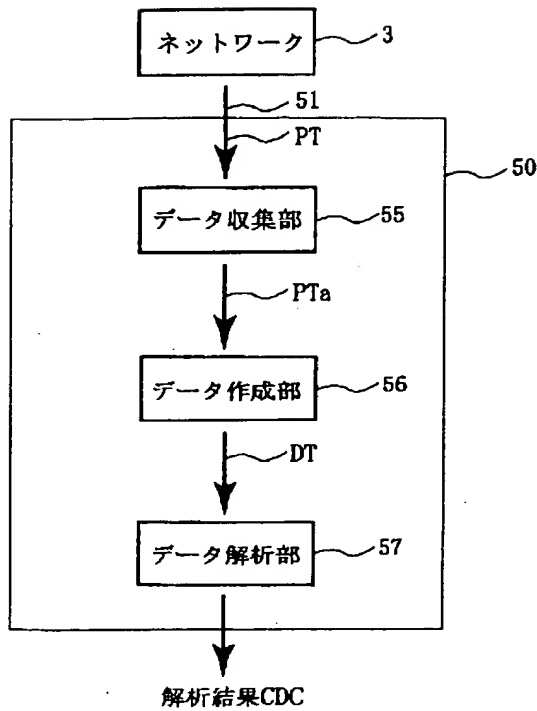
【図2】



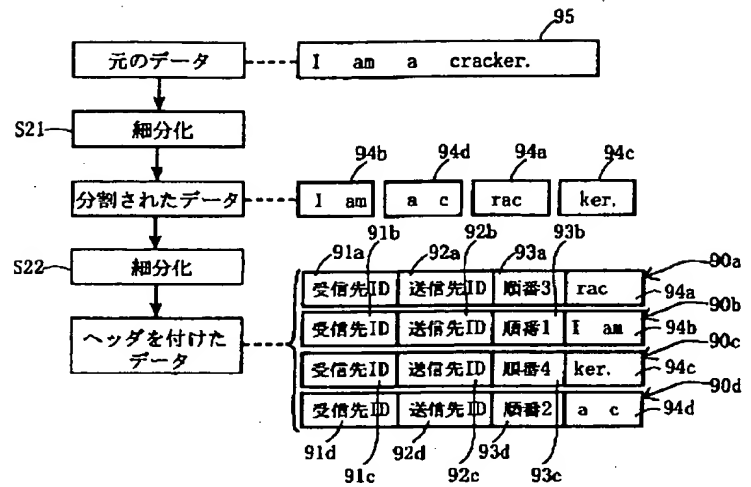
【図5】



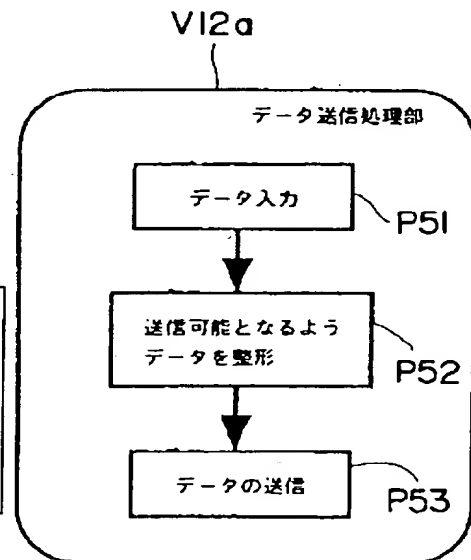
【図 3】



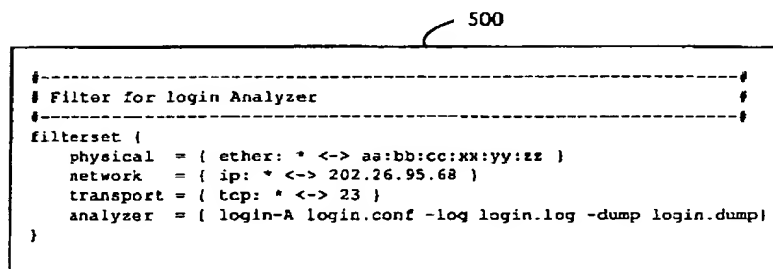
【図 7】



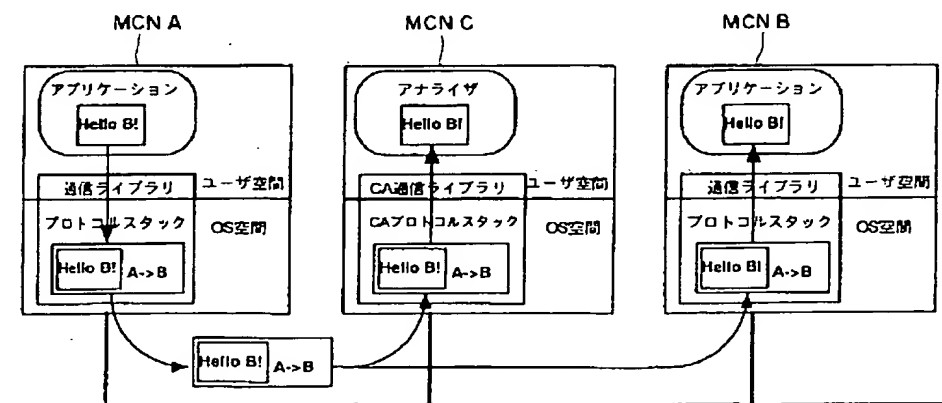
【図 17】



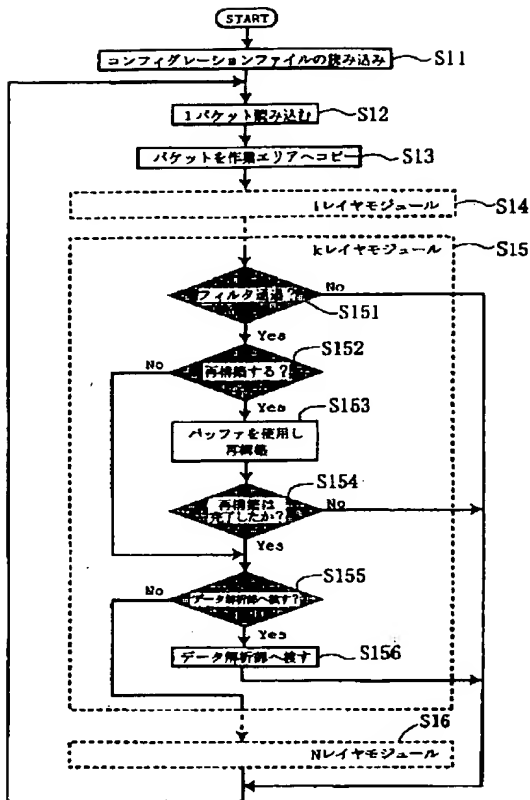
【図 4】



【図 18】



【図6】



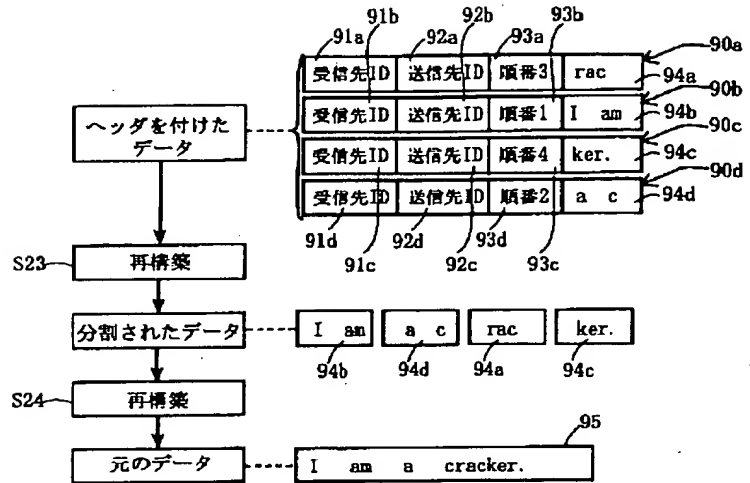
【図9】

```

----- Cracking Data contents -----
Date       : Dec 26 13:28:00 1996
Server Address : 202.26.95.68:23
Client Address : 202.26.95.69:1024
Defhost Name  : host1
Login Times   : Threshold User = 0 Threshold System = 3
Easy Password Used : NO

----- Command Details -----
S :
S : Welcome to mail.nal.go.jp (tttyp4)
S :
S : login:
C : root
S : Password:
C : roots
S : Login incorrect
S : login:
C : root
S : Password:
C : root
  
```

【図8】



【図22】

(a) 本発明の通信ライブラリ

```

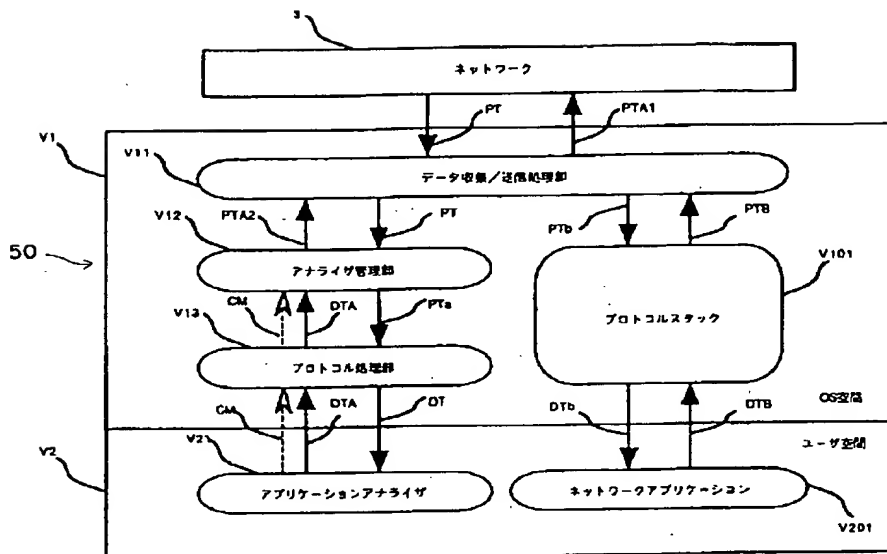
int fd,new_fd;
fd=socket();
bind(fd,監視対象のアドレスルール);
while(1){
    new_fd=accept(fd);
    if(fork() == 0){
        close(fd);
        new_fdを使ってサーバ〜クライアントのデータ通信を解析する
    }
    close(new_fd);
}
  
```

(b) 従来の通信ライブラリ (サーバの場合)

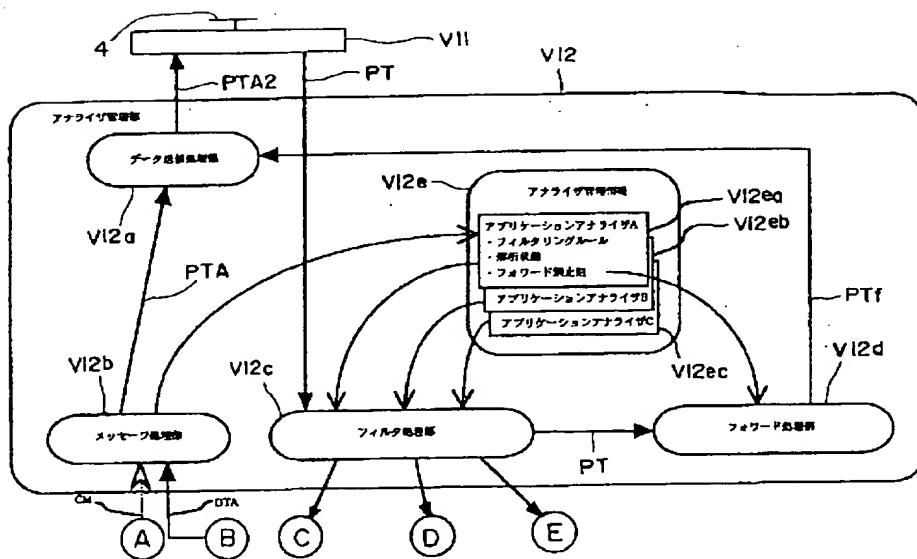
```

int fd,new_fd;
fd=socket();
bind(fd,開設するポート);
while(1){
    new_fd=accept(fd);
    if(fork() == 0){
        close(fd);
        new_fdを使ってクライアントとデータ通信を行う
    }
    close(new_fd);
}
  
```

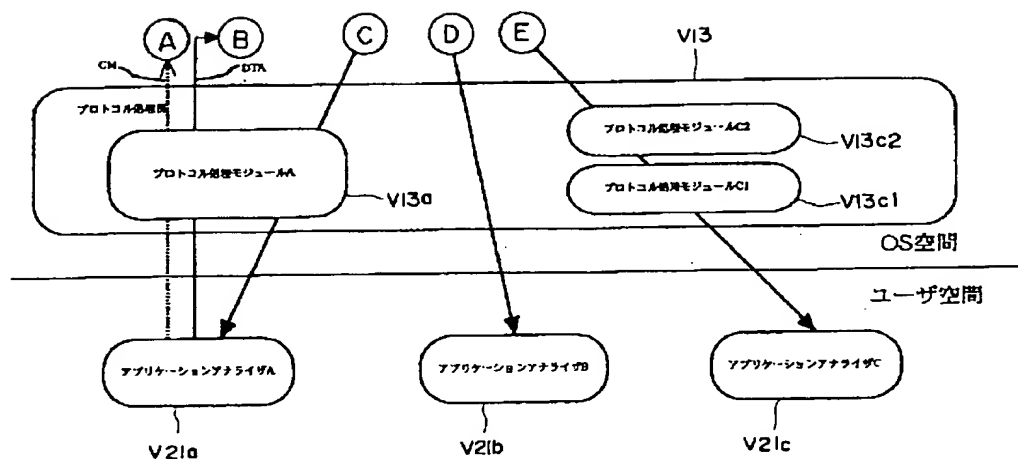
【図 10】



【☒ 1 1】



【図12】



【図13】

```

struct ca_filter {
    char    dir;                /* direction */

    /*-----*/
    /* datalink layer */
    /*-----*/
    char    dlk_id;
    union {
        struct ether_filter    ether;
    } dlk_f;

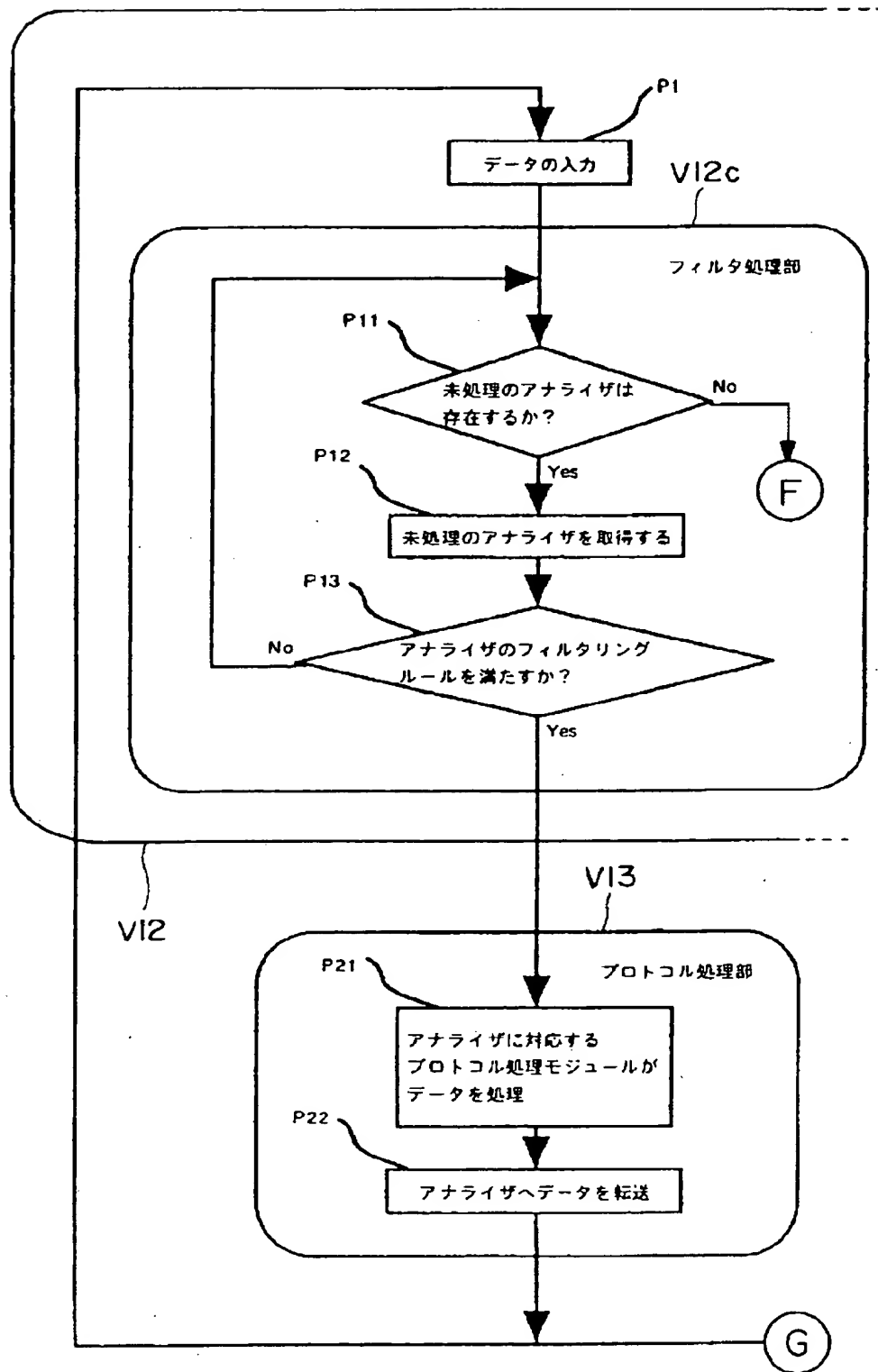
    /*-----*/
    /* network layer */
    /*-----*/
    char    net_id;
    union {
        struct ip_filter        ip;
        struct arp_filter        arp;
    } net_f;

    /*-----*/
    /* transport layer */
    /*-----*/
    char    tra_id;
    union {
        struct tcp_filter        tcp;
        struct udp_filter        udp;
    } tra_f;
};

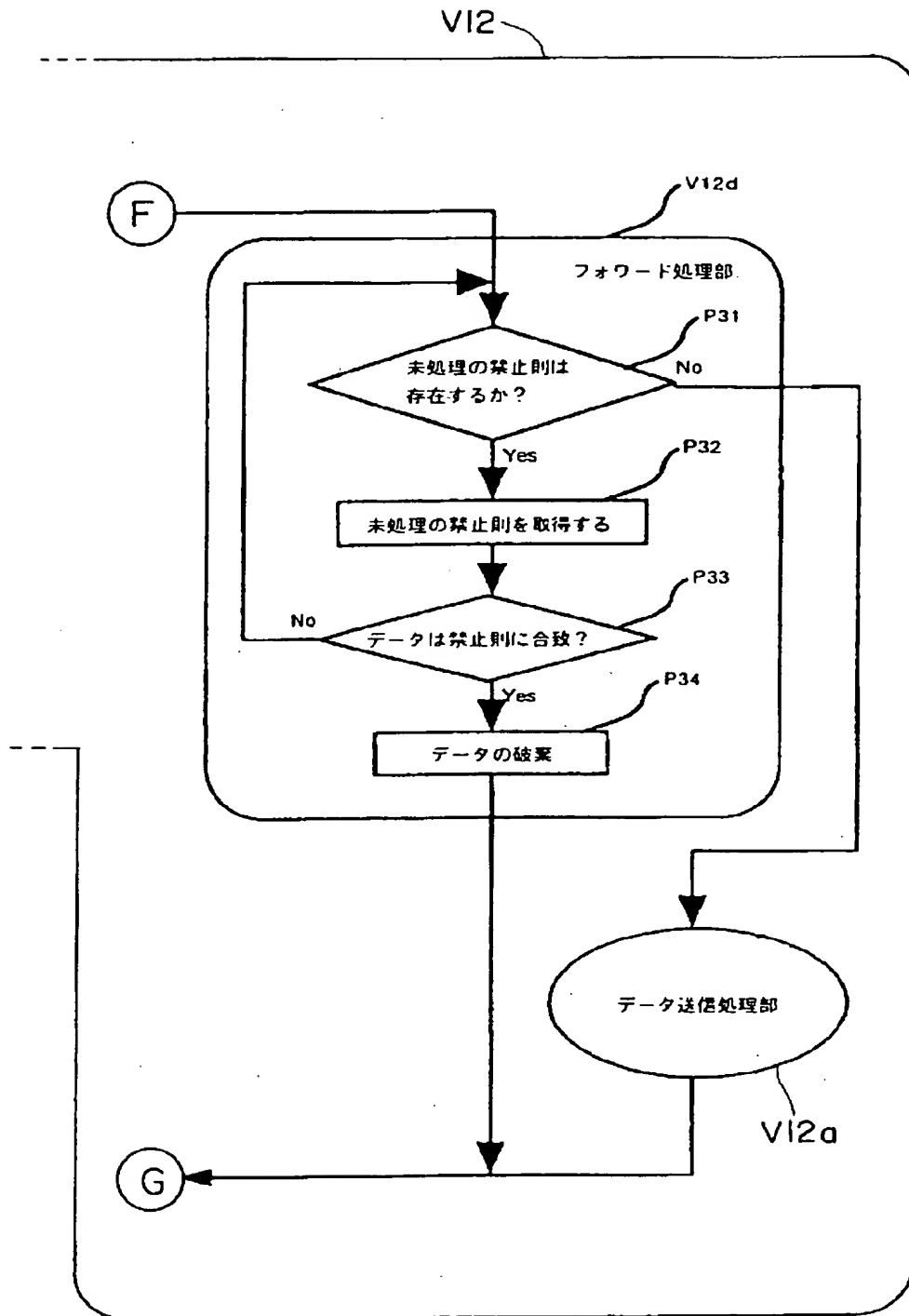
struct ca_endpoint {
    queue_t*    qptr;           /* points to AA */
    short        stat;           /* status */
    struct ca_filter    filter;  /* filtering rule */
    struct ca_filter    nf_filter; /* no-forward filtering rule */
    int          id;             /* identifier */
    int          fd;             /* file descriptor */
    int          bufcid;
    struct ca_endpoint*    ca_e;  /* used in accept */
    struct ca_eopt    opt;        /* option */
};

```

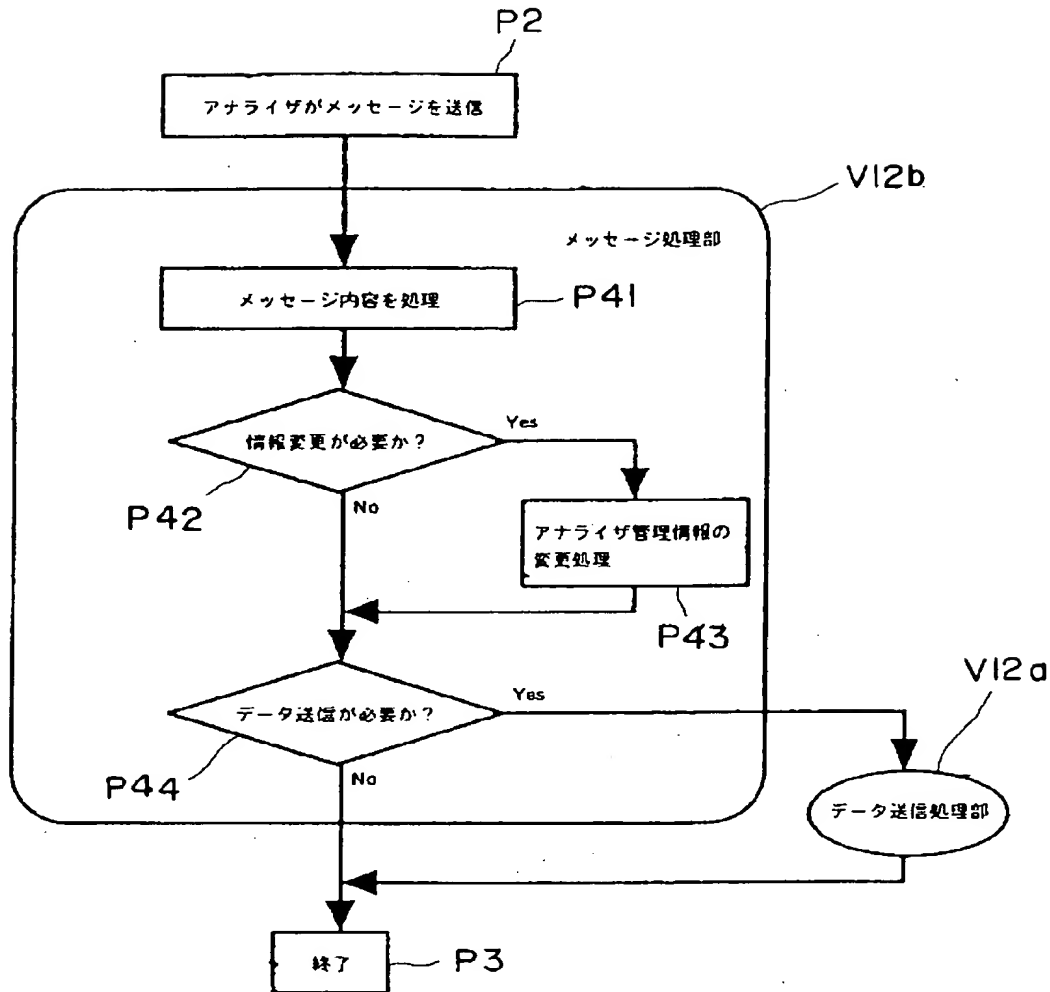

【図14】



【図15】



【図16】



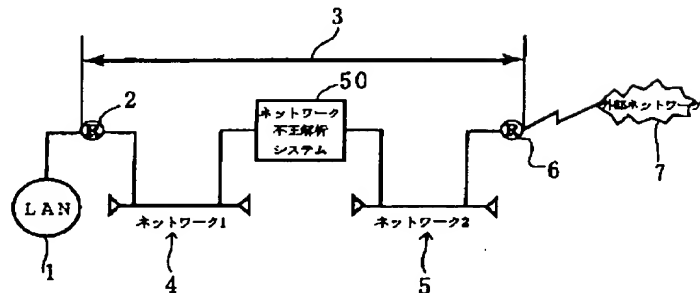
【図23】

```

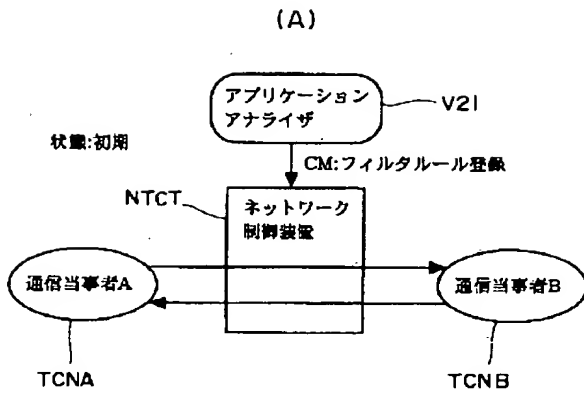
int fd, new_fd;
fd=socket();
bind(fd, 監視対象のアドレスルール);
while(1){
  new_fd=accept(fd);
  if(fork() == 0){
    close(fd);
    hijack(new_fd);
    while(1){
      new_fdを使ってサーバ〜クライアントのデータ通信を解析する
      write(new_fd, 送信するデータ, 送信先);
    }
  }
  close(new_fd);
}

```

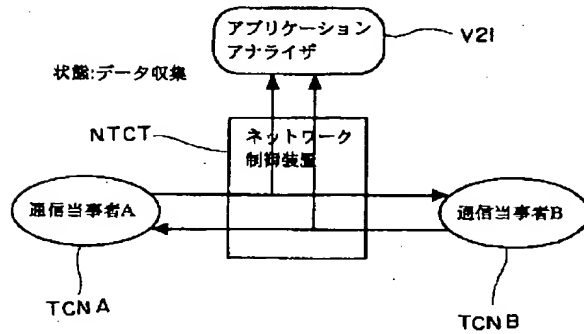
【図27】



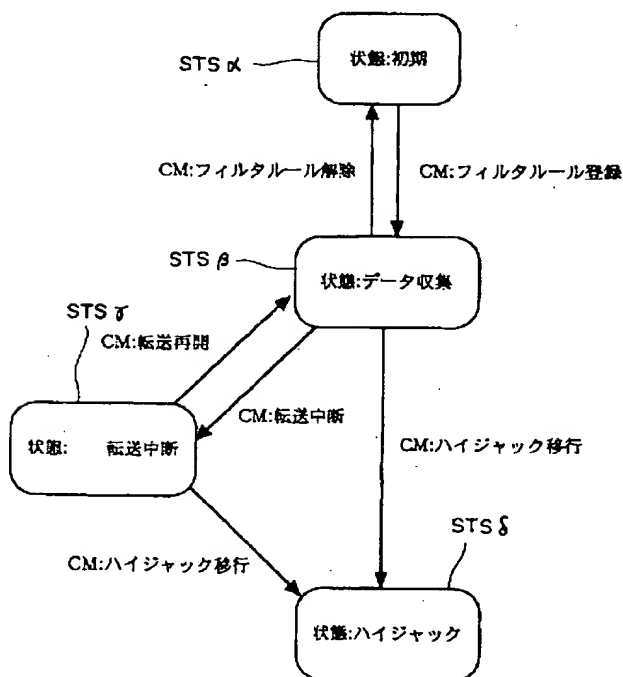
【図19】



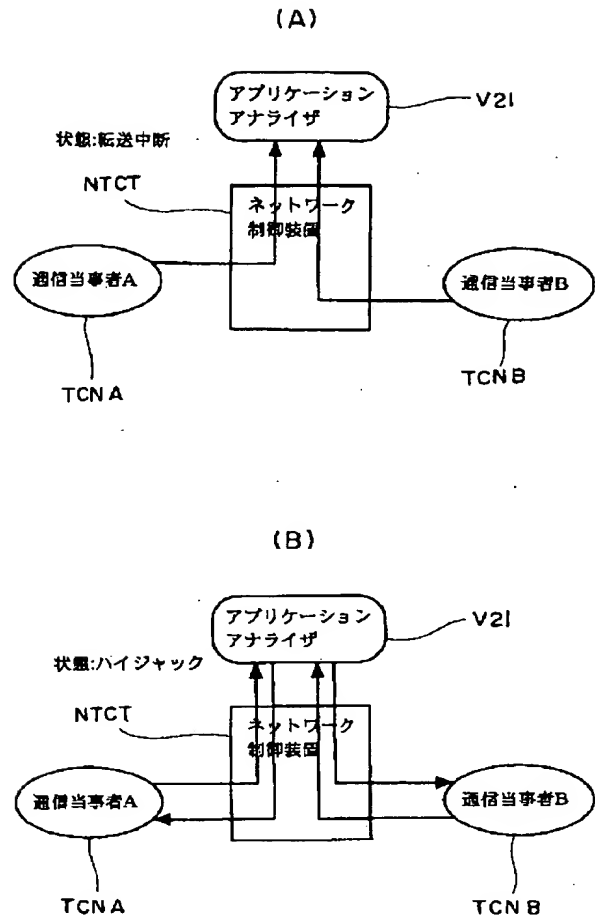
(B)



【図21】



【図20】



【図24】

```

-----
|                                     |
|                               START |
|                                     |
-----

ETHER: 00:80:17:88:8d:ca >> 00:a0:24:32:18:0d
IP: 202.26.75.116 >> 202.26.75.71
TCP: 40508 >> 23

[ src << (DST) ](id:4553)<len:7> Mon Aug 31 09:58:13 1998
...
BSD: BSD/OS 3.0 (netlog.nal.go.jp) (ttyp0)...
...
...login:
[ (SRC) >> dst ](id:4553)<len:10> Mon Aug 31 09:58:13 1998
...
[ src << (DST) ](id:4553)<len:0> Mon Aug 31 09:58:14 1998
[ (SRC) >> dst ](id:4553)<len:1> Mon Aug 31 09:58:18 1998
[ src << (DST) ](id:4553)<len:1> Mon Aug 31 09:58:18 1998
[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:18 1998
[ (SRC) >> dst ](id:4553)<len:1> Mon Aug 31 09:58:22 1998
[ src << (DST) ](id:4553)<len:1> Mon Aug 31 09:58:22 1998
[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:23 1998
[ (SRC) >> dst ](id:4553)<len:1> Mon Aug 31 09:58:23 1998
[ src << (DST) ](id:4553)<len:1> Mon Aug 31 09:58:23 1998
[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:23 1998
[ (SRC) >> dst ](id:4553)<len:1> Mon Aug 31 09:58:24 1998
[ src << (DST) ](id:4553)<len:10> Mon Aug 31 09:58:24 1998
[ (SRC) >> dst ](id:4553)<len:10> Mon Aug 31 09:58:24 1998
[ src << (DST) ](id:4553)<len:4> Mon Aug 31 09:58:24 1998

```

【図25】

```

[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:24 1998
[ src << (DST) ](id:4553)<len:12> Mon Aug 31 09:58:24 1998
...Password:
[ (SRC) >> dst ](id:4553)<len:3> Mon Aug 31 09:58:24 1998
...
[ src << (DST) ](id:4553)<len:0> Mon Aug 31 09:58:24 1998
[ (SRC) >> dst ](id:4553)<len:7> Mon Aug 31 09:58:32 1998
root01
[ src << (DST) ](id:4553)<len:2> Mon Aug 31 09:58:32 1998
[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:32 1998
[ src << (DST) ](id:4553)<len:31> Mon Aug 31 09:58:32 1998
...
..login incorrect.

login:
[ (SRC) >> dst ](id:4553)<len:7> Mon Aug 31 09:58:32 1998
...
[ src << (DST) ](id:4553)<len:0> Mon Aug 31 09:58:32 1998
[ (SRC) >> dst ](id:4553)<len:1> Mon Aug 31 09:58:37 1998
[ src << (DST) ](id:4553)<len:4> Mon Aug 31 09:58:37 1998
...
[ src << (DST) ](id:4553)<len:10> Mon Aug 31 09:58:37 1998
[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:37 1998
[ (SRC) >> dst ](id:4553)<len:0> Mon Aug 31 09:58:37 1998
[ src << (DST) ](id:4553)<len:0> Mon Aug 31 09:58:37 1998

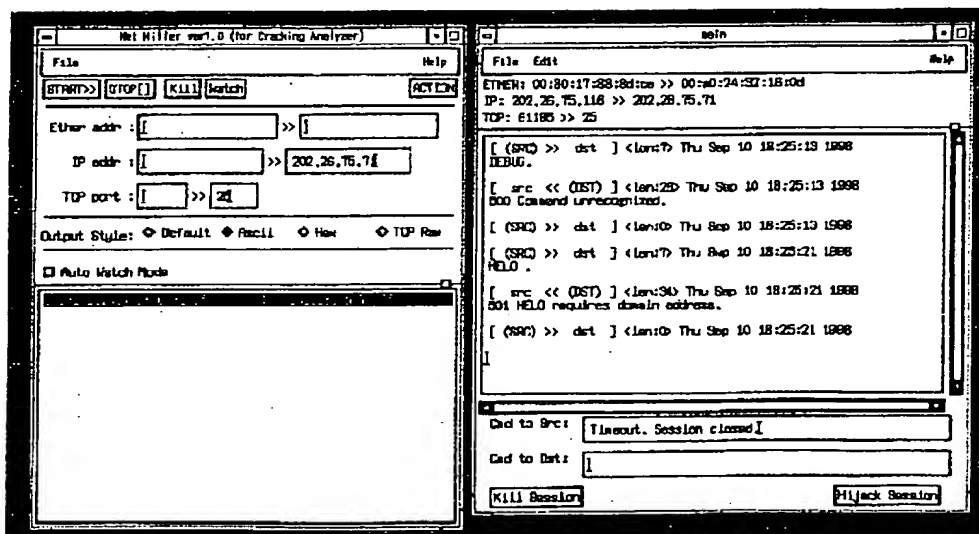
```

```

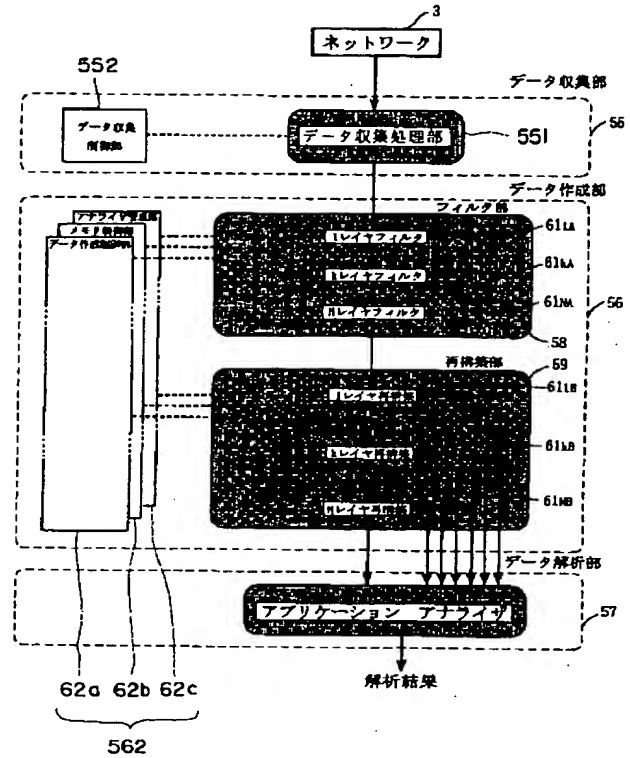
-----
|                                     |
|                               END   |
|                                     |
-----

```

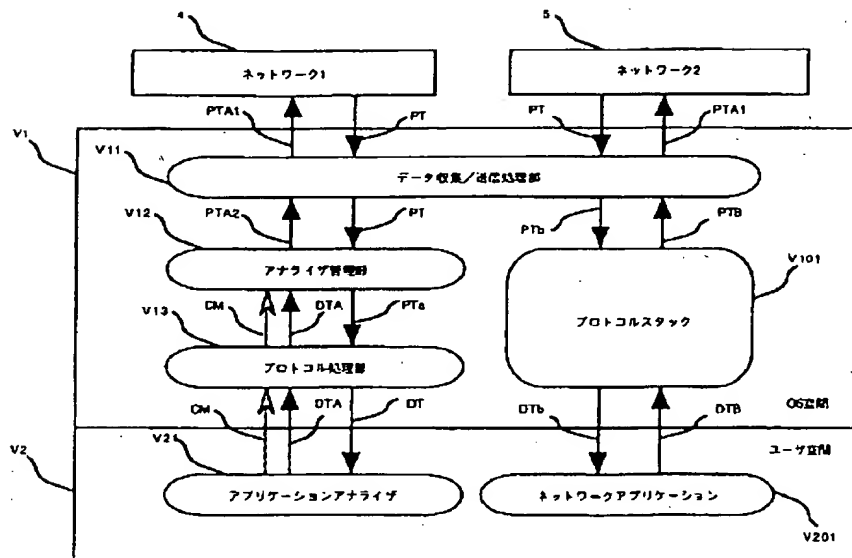
【図26】



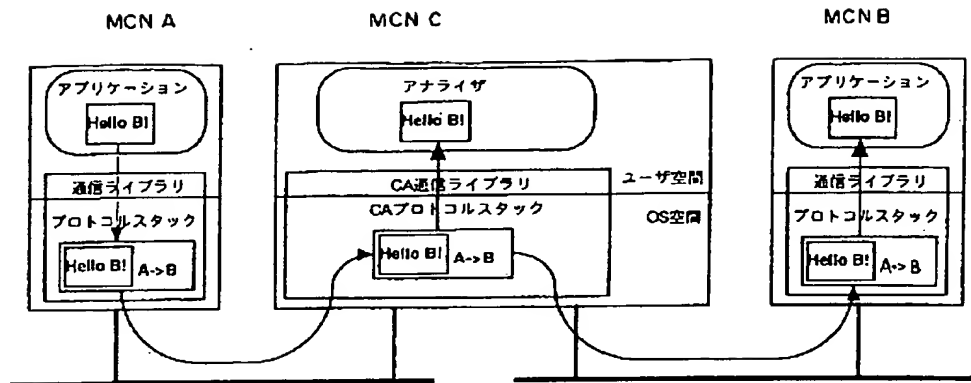
【图 29】



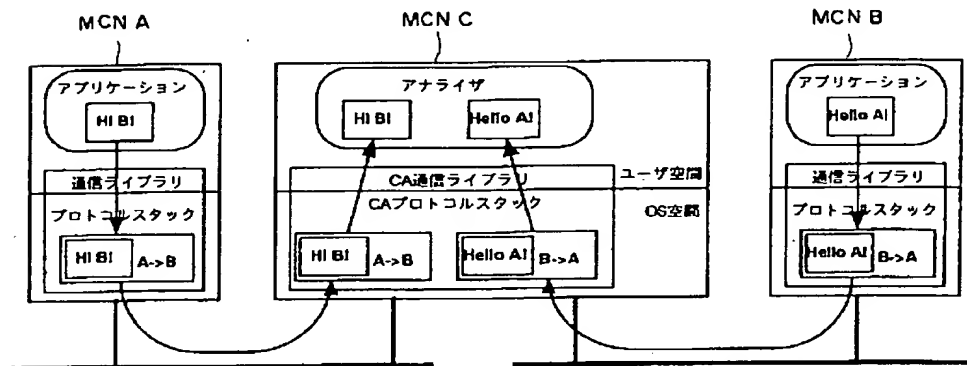
【図 30】



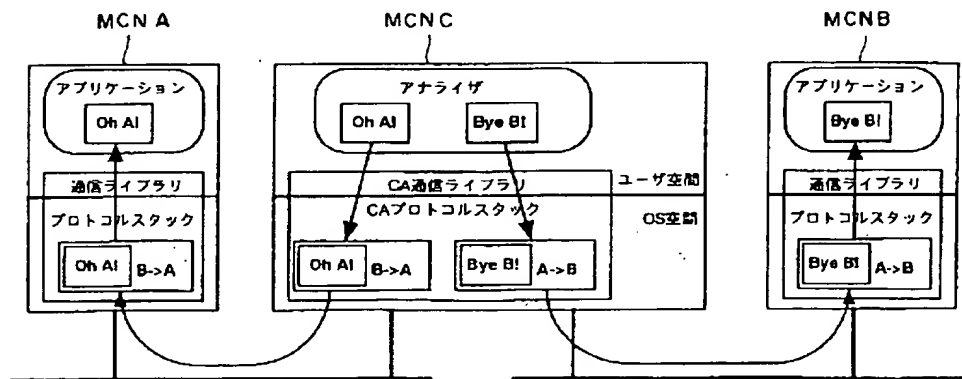
【図31】



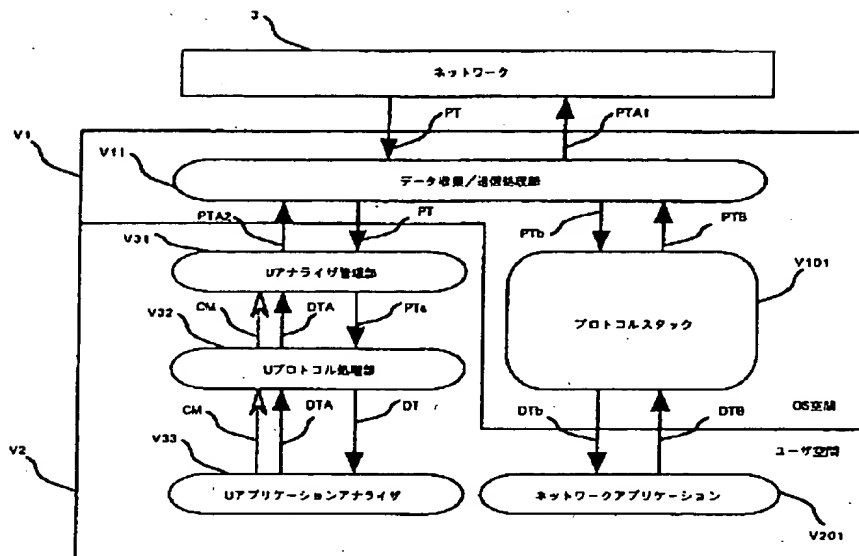
【図32】



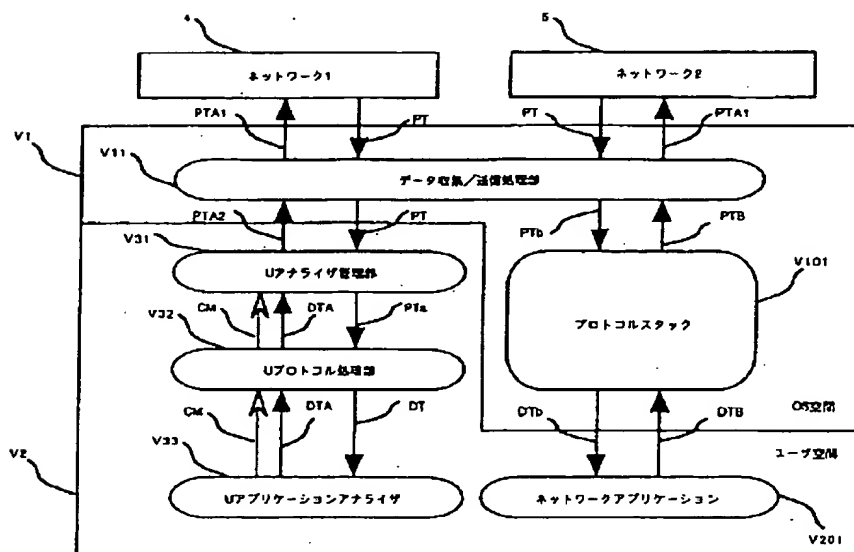
【図33】



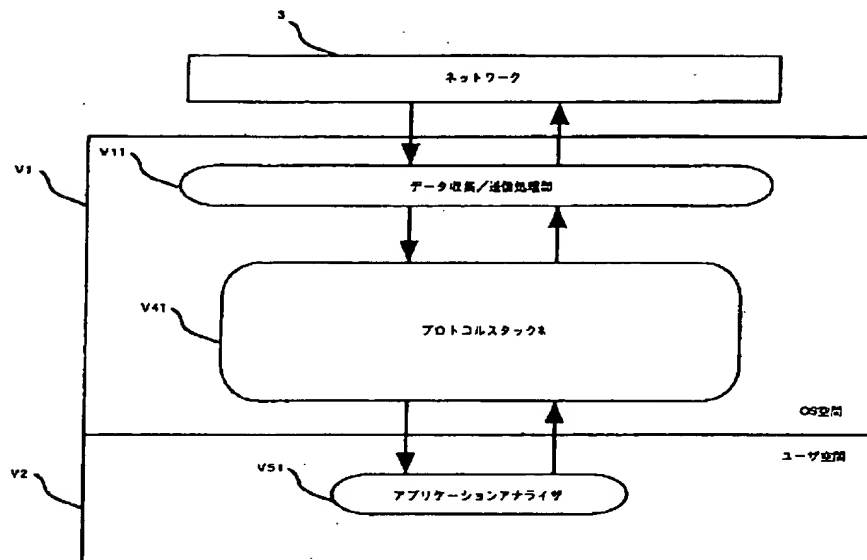
【図34】



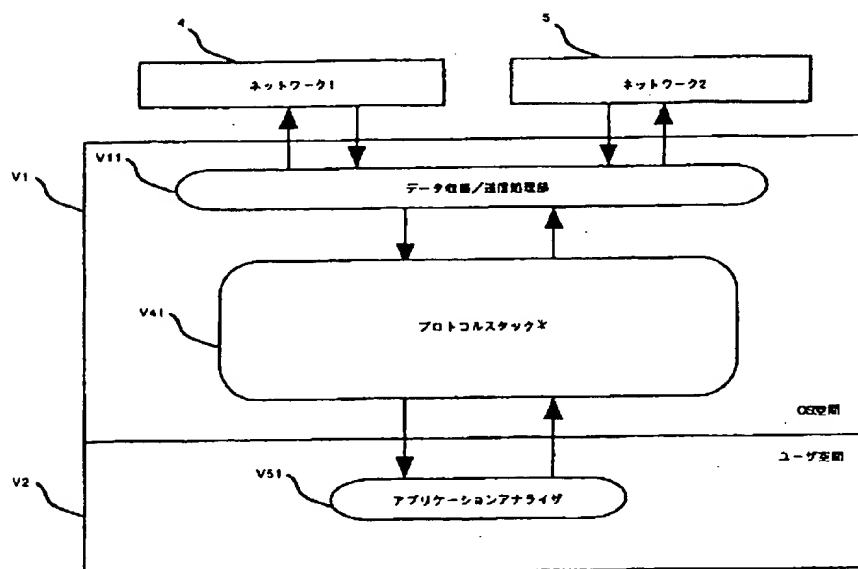
【図35】



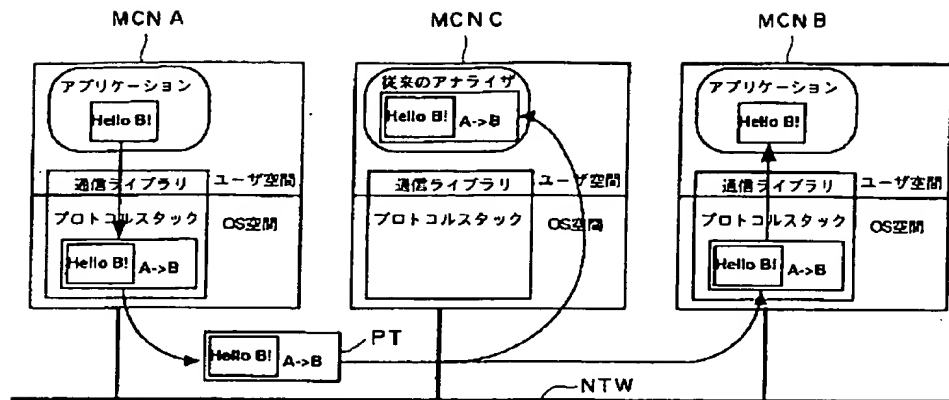
【図36】



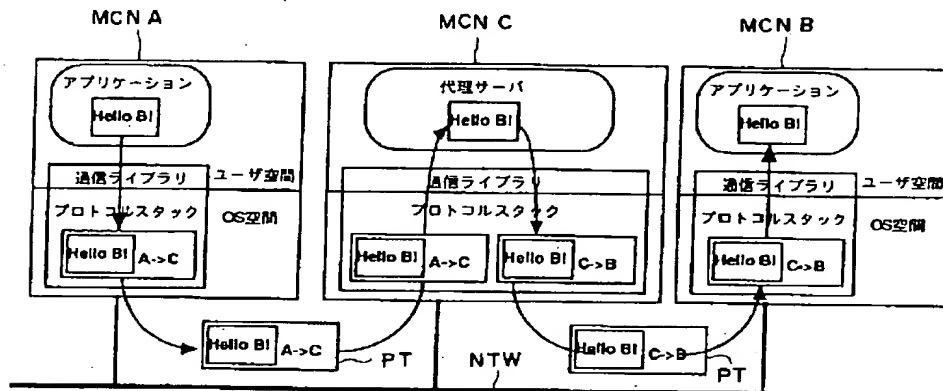
【図37】



【図38】



【図39】



フロントページの続き

(72)発明者 藤田 直行
東京都小金井市前原町4-17-28

Fターム(参考) 5B089 GA01 GB02 HB11 JB19 KB03
KB13 KC47 KE01 MC08
5K030 GA16 HA08 MC07